

Testiranje performansi sistema pametne kuće

SV8/2020 Bojan Mijanović

SV9/2020 Vukašin Bogdanović

SV47/2020 Jovan Jokić

Sadržaj

Uvod	3
Student 1.....	3
Prijavljivanje korisnika.....	3
Registracija korisnika.....	4
Dobavljanje podataka o zakazanim terminima veš mašine	5
Kreiranje zakazanih termina za rad klima uređaja	6
Deljenje uređaja sa drugim korisnicima.....	7
Dobavljanje podataka o istoriji vrednosti ambijentalnog senzora	8
Promena režima rada veš mašine	9
Promena temperature rada klima uređaja	11
Dodavanje novog administratora.....	12
Promena šifre korisnika.....	13
Student 2.....	15
Registracija pametne nekretnine	15
Provera dostupnosti uređaja.....	16
Dobavljanje podataka o istoriji akcija kapije	17
Dobavljanje podataka o lampi.....	18
Otvaranje/Zatvaranje kapije.....	19
Dodavanje verifikovane tablice za kapiju	20
Brisanje verifikovane tablice za kapiju	21
Dobavljanje svih nekretnina jednog korisnika	22
Kreiranje zakazanih termina za rad prskalice	23
Promena režima rada lampe	24
Student 3.....	26
Dobavljanje podataka o bateriji u pametnoj kući	26
Paljenje/Gašenje Solarnih Panela	27
Registracija uređaja.....	28
Proizvodnja/Potrošnja svih uređaja unutar nekretnine	29
Dodavanje automobila na punjač	30
Dobavljanje uređaja u okviru kuće.....	32
Dobavljanje podataka o EV punjaču u pametnoj kući.....	33
Dobavljanje potrošnje po gradu.....	34
Dobavljanje svih gradova	35
Dobavljanje istorijskih podataka o akcijama nad solarnim panelom.....	37
Testiranje performansi sistema	39

Uvod

Za sprovođenje testova opterećenja koristićemo Locust alat. Slali smo nasumične podatke kako bismo simulirali stvarne scenarije. Planirani su testovi sa 100-1000 korisnika tokom 1 minute, postepeno uvodeći rast od 30 korisnika po sekundi. Pratimo metrike poput procenta uspešnosti zahteva, broja zahteva u sekundi i vremena potrebnog za dobijanje odgovora od strane servera. Svi testovi će biti testirani nad laptop uređajem sa procesorom i3 desete generacije, integrisanom grafičkom karticom i 8gb RAM memorije.

Student 1

Prijavljivanje korisnika

Testiranje prijavljivanja korisnika je jedno od najbitnijih funkcionalnosti sistema. Test sa sastoji od jednostavnog HTTP zahteva koji će pokušati da identifikuje da li su kredencijali korisnika postojeći u sistemu. Nakon testiranja sistema dobili smo uspešnu identifikaciju u 89% slučajeva. Najveće zagušenje sistema je predstavljala konekcija za bazom koja se nalazila u docker container-u.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/User/login	405	47	20679	1873	39199	461	9.4	1.1
	Aggregated	405	47	20679	1873	39199	461	9.4	1.1

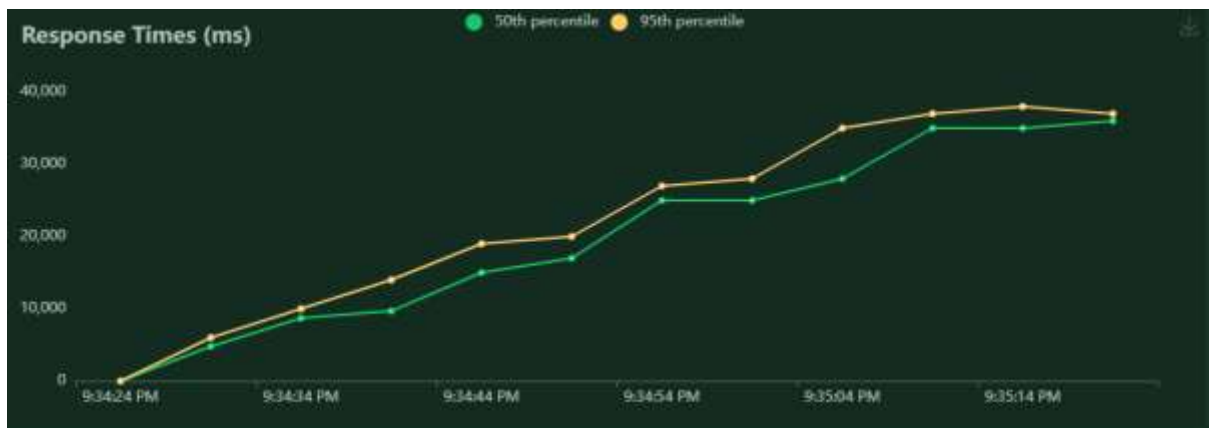
Statistika zahteva tokom login operacije

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/User/login	24000	25000	26000	28000	35000	36000	37000	39000
	Aggregated	24000	25000	26000	28000	35000	36000	37000	39000

Statistika odziva sistema tokom login operacije



Grafik zahteva i padova u sekundi tokom vremena tokom login operacije



Grafik vremena odziva sistema tokom vremena tokom login operacije

Registracija korisnika

Testiranje registracije korisnika je funkcionalnost neophodna za rad celokupnog sistema. Test sa sastoji od jednostavnog HTTP POST zahteva koji će slati razne podatke o potencijalnom korisniku kao što su ime i prezime, kredencijali za prijavu, email adresa i fajl koji predstavlja profilnu sliku korisnika. Nakon testiranja sistema dobili smo uspešnu identifikaciju u 100% slučajeva. Najveće zagušenje sistema je predstavljala obrada velikih slika i zapisivanje istih u perzistentno skladište. Jedan od načina za unapređenje ove funkcionalnosti je ograničavanje/kompersija profilne slike korisnika čime bi obrada velikih slika bila mnogo brža.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/User/register	419	0	9658	449	25205	341	7.4	0.0
	Aggregated	419	0	9658	449	25205	341	7.4	0.0

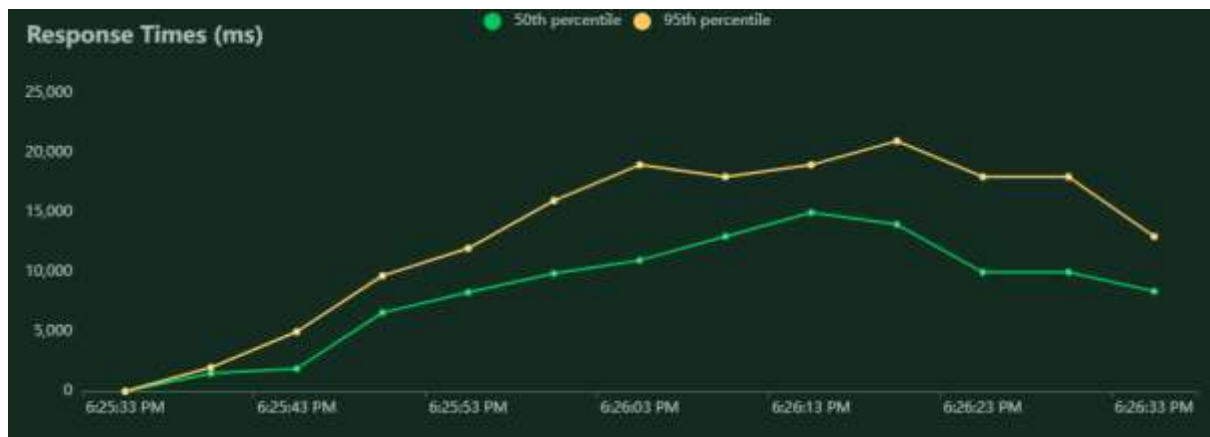
Statistika zahteva tokom registracije korisnika

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/User/register	9700	11000	12000	13000	16000	18000	21000	25000
	Aggregated	9700	11000	12000	13000	16000	18000	21000	25000

Statistika odziva sistema tokom registracije korisnika



Grafik zahteva i padova u sekundi tokom vremena tokom registracije korisnika



Grafik vremena odziva sistema tokom vremena tokom registracije korisnika

Dobavljanje podataka o zakazanim terminima već mašine

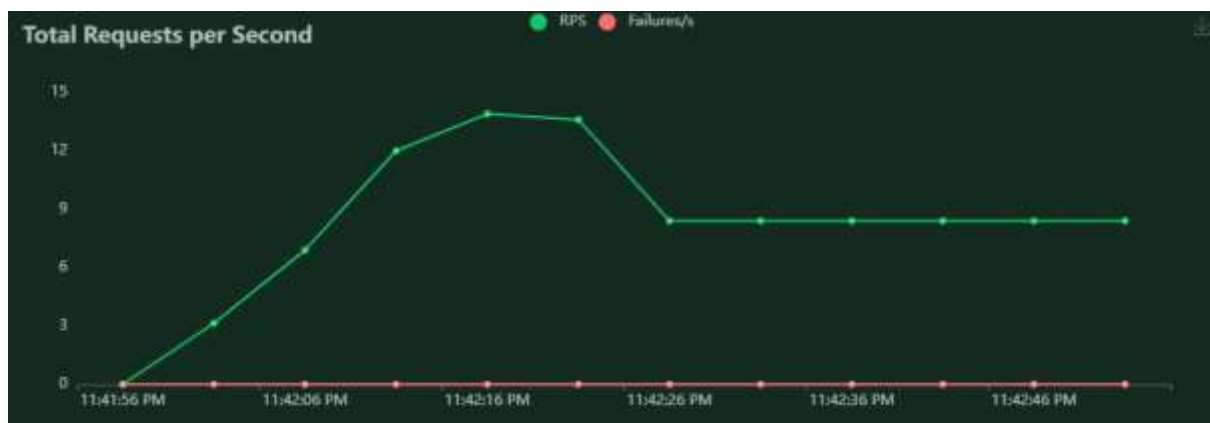
Test se sastoji od prijavljivanja korisnika i slanja GET zahteva sa identifikacionim brojem uređaja ka serveru koji mora da proveri dozvole korisnika, postojanost uređaja i dobavi podatke iz perzistentnog skladišta. Nakon testiranja sistema dobili smo uspešnu identifikaciju u 100% slučajeva. Najveće opterećenje predstavlja opterećenje ka PostgreSQL bazi podataka.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/User/login	228	1	13173	1881	25683	39	7.2	0.0
GET	/api/WashingMachine/Get?id=050b583c-1581-4969-9d10-f5607b7d3809	55	0	8440	41	14504	360	1.7	0.0
Aggregated		283	1	12253	41	25683	101	8.9	0.0

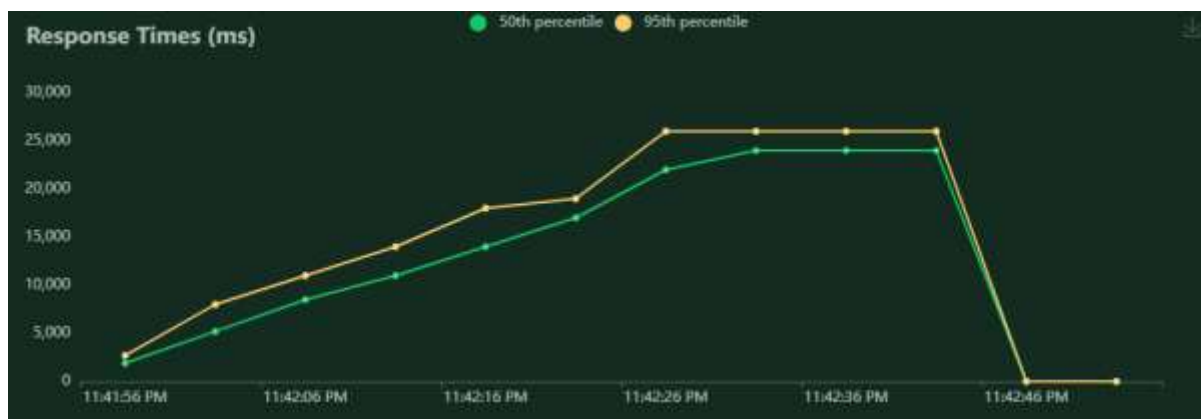
Statistika zahteva tokom dobavljanja podataka o terminima rada već mašine

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/User/login	13000	14000	16000	18000	19000	23000	25000	26000
GET	/api/WashingMachine/Get?id=050b583c-1581-4969-9d10-f5607b7d3809	9800	10000	12000	12000	14000	14000	15000	15000
Aggregated		12000	13000	15000	17000	18000	23000	25000	26000

Statistika odziva sistema tokom dobavljanja podataka o terminima rada već mašine



Grafik zahteva i padova u sekundi tokom vremena tokom dobavljanja podataka o terminima rada već mašine



Grafik vremena odziva sistema tokom vremena tokom dobavljanja podataka o terminima rada već mašine

Kreiranje zakazanih termina za rad klima uređaja

Test se sastoji od jednostavnog HTTP PUT zahteva koji će slati podatke o terminu kada je neophodno da se klima pali, koliko dugo da radi u kom režimu i na kojoj temperaturi. Najveći izazov ove funkcionalnosti je izmena i objavljivanje novih akcija što opterećuje mqtt komunikaciju kao i sam server. Nakon testiranja sistema dobili smo uspešnu identifikaciju u 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/AirConditioner/AddScheduledWork	286	0	14242	132	31484	0	5.0	0.0
POST	/api/User/login	100	0	6517	673	26824	23	1.7	0.0
Aggregated		386	0	12240	132	31484	5	6.7	0.0

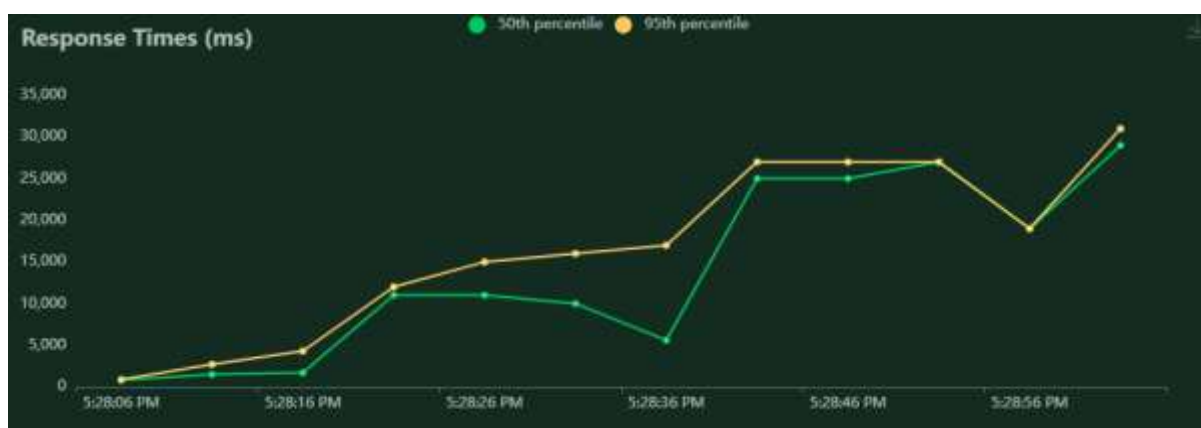
Statistika zahteva tokom kreiranja zakazanih termina za rad klima uređaja

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/AirConditioner/AddScheduledWork	11000	14000	26000	30000	30000	31000	31000	31000
POST	/api/User/login	1700	4300	10000	15000	16000	23000	27000	27000
Aggregated		10000	12000	16000	29000	30000	31000	31000	31000

Statistika odziva sistema tokom kreiranja zakazanih termina za rad klima uređaja



Grafik zahteva i padova u sekundi tokom vremena tokom kreiranja zakazanih termina za rad klima uređaja



Grafik vremena odziva sistema tokom vremena tokom kreiranja zakazanih termina za rad klima uređaja

Deljenje uređaja sa drugim korisnicima

Test se sastoji od prijave korisnika i jednostavnog HTTP PUT zahteva koji će ka serveru slati podatke o tome koji uređaj i kuća će postati dostupni korisniku čiju email adresu šaljemo. Najveći izazov ovog test slučaja je ponovno kreiranje redis keša koje oduzima dosta procesorskog vremena, ali i optimizuje situaciju dobavljanja korisnikovih uređaja. Nakon testiranja sistema dobili smo uspešnu identifikaciju u 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
PUT	/api/SmartHome/AddPermission	151	0	20677	5453	29026	0	2.5	0.0
POST	/api/User/login	100	0	6795	1191	22060	23	1.7	0.0
Aggregated		251	0	15146	1191	29026	9	4.2	0.0

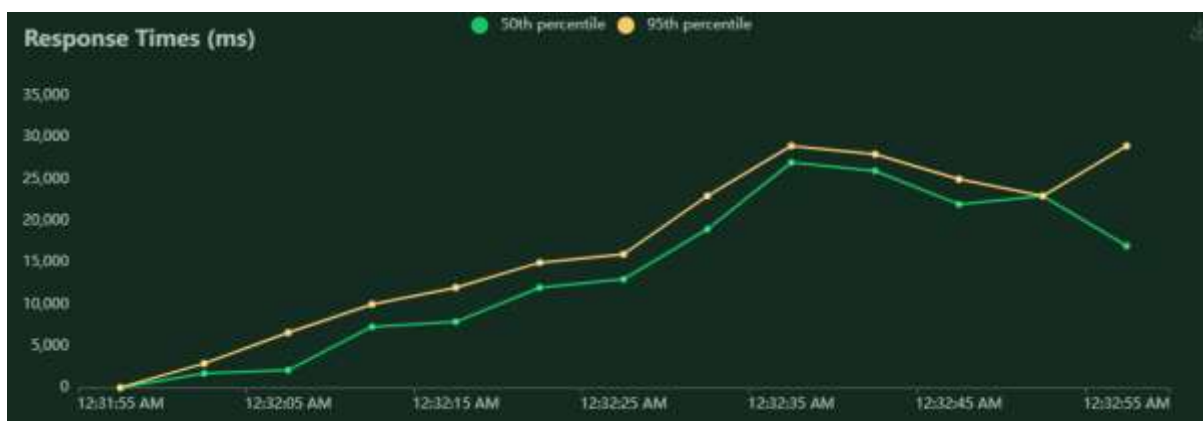
Statistika zahteva tokom deljenja uređaja sa drugim korisnicima

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
PUT	/api/SmartHome/AddPermission	23000	25000	26000	27000	28000	28000	29000	29000
POST	/api/User/login	6700	7200	7800	11000	13000	14000	22000	22000
	Aggregated	13000	19000	23000	26000	27000	28000	29000	29000

Statistika odziva sistema tokom *deljenja uređaja sa drugim korisnicima*



Grafik zahteva i padova u sekundi tokom vremena tokom *deljenja uređaja sa drugim korisnicima*



Grafik vremena odziva sistema tokom vremena tokom *deljenja uređaja sa drugim korisnicima*

Dobavljanje podataka o istoriji vrednosti ambijentalnog senzora

Testiranje dobavljanja podataka o ambijentalnom senzoru je neophodna operacija za praćenje rada ambijentalnog senzora u pametnoj kući. Test se sastoji od prijavljivanja korisnika, nakon čega se šalje GET zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju ambijentalnog senzora. . Najveći izazov ovog zahteva predstavlja potencijalno dobavljanje velike količine podataka iz time-series Influx baze podataka. Ovaj problem je rešen uvođenjem određenog opsega iz kog se dobavljaju podaci, kao i ograničavanjem maksimalnog opsega na 30 dana. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/AmbientSensor/GetHistoricalData?id=4f84c687-8338-4200-958e-0ae99a38c1ad&from=2023-01-01%2000:00:00&to=2023-12-31%2000:00:00	2684	0	44	19	602	1424	44.8	0.0
POST	/api/User/login	100	0	1359	955	3679	23	1.7	0.0
Aggregated		2784	0	91	19	3679	1373	46.4	0.0

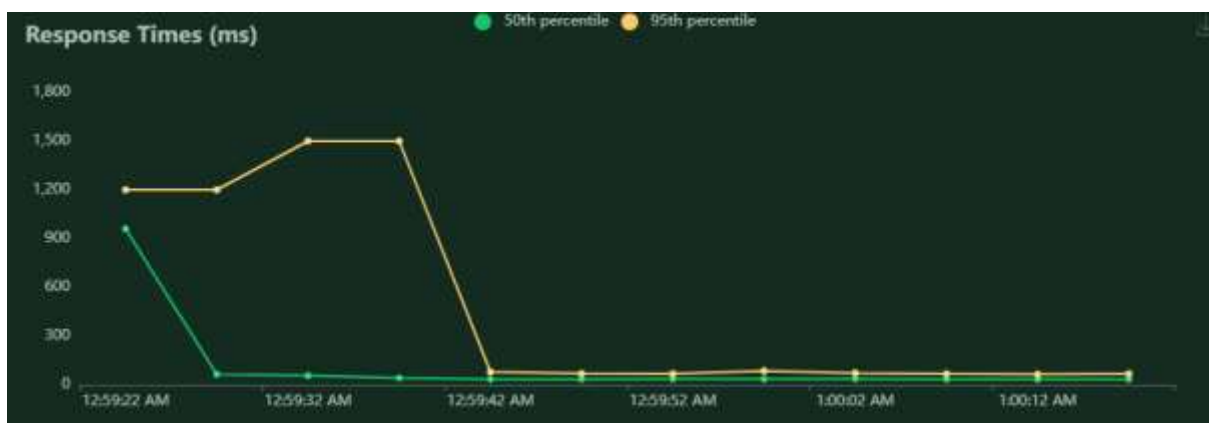
Statistika zahteva tokom dobavljanja istorijskih podataka o ambijentalnom senzoru

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	/api/AmbientSensor/GetHistoricalData?id=4f84c687-8338-4200-958e-0ae99a38c1ad&from=2023-01-01%2000:00:00&to=2023-12-31%2000:00:00	35	38	44	53	67	83	220	600
POST	/api/User/login	1200	1200	1400	1600	2000	2700	3700	3700
Aggregated		35	39	45	56	75	170	1400	3700

Statistika odziva sistema tokom dobavljanja istorijskih podataka o ambijentalnom senzoru



Grafik zahteva i padova u sekundi tokom dobavljanja istorijskih podataka o ambijentalnom senzoru



Grafik vremena odziva sistema tokom dobavljanja istorijskih podataka o ambijentalnom senzoru

Promena režima rada veš mašine

Testiranje promene režima rada veš mašine je neophodna operacija za moduliranje rada veš mašine u pametnoj kući. Test se sastoji od prijavljivanja korisnika, nakon čega se šalje PUT zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju veš mašine i novom režimu rada veš mašine. Najveći

izazov ovog zahteva predstavlja izmena podataka u PostgreSQL bazi podataka i Influx bazi podataka. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/User/login	100	0	10259	640	30438	23	1.9	0.0
PUT	/api/WashingMachine/ChangeMode?id=39c9d210-d914-48a1-9eb9-6dde382b8e34&mode=AntiAllergy	102	0	10664	220	28187	0	2.0	0.0
PUT	/api/WashingMachine/ChangeMode?id=39c9d210-d914-48a1-9eb9-6dde382b8e34&mode=Mixed%20wash	105	0	9686	109	28509	0	2.0	0.0
PUT	/api/WashingMachine/ChangeMode?id=39c9d210-d914-48a1-9eb9-6dde382b8e34&mode=White%20wash	89	0	10963	165	28302	0	1.7	0.0
Aggregated		396	0	10370	109	30438	5	7.6	0.0

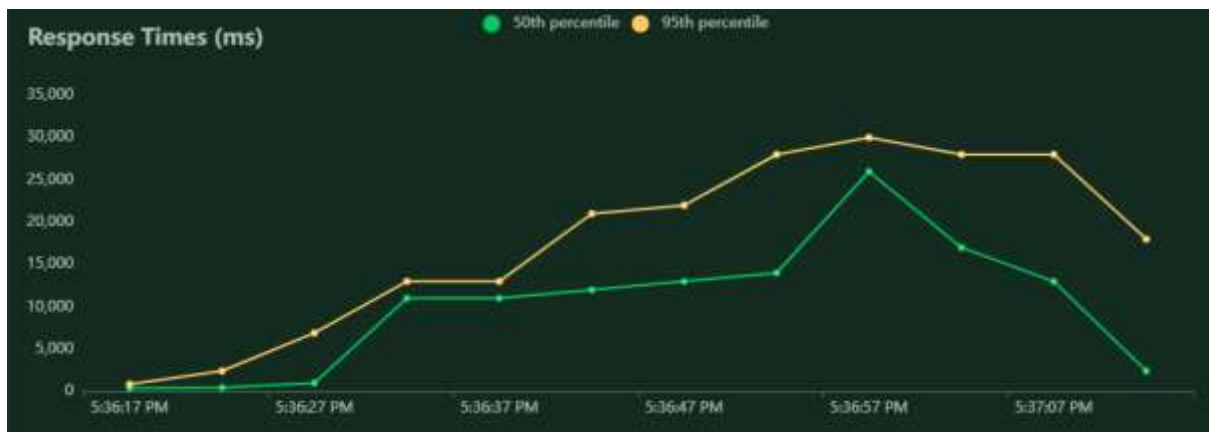
Statistika zahteva tokom promene režima rada veš mašine

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/User/login	11000	12000	13000	22000	24000	28000	30000	30000
PUT	/api/WashingMachine/ChangeMode?id=39c9d210-d914-48a1-9eb9-6dde382b8e34&mode=AntiAllergy	11000	12000	14000	17000	26000	28000	28000	26000
PUT	/api/WashingMachine/ChangeMode?id=39c9d210-d914-48a1-9eb9-6dde382b8e34&mode=Mixed%20wash	10000	11000	13000	18000	22000	28000	28000	29000
PUT	/api/WashingMachine/ChangeMode?id=39c9d210-d914-48a1-9eb9-6dde382b8e34&mode=White%20wash	11000	13000	15000	17000	24000	26000	28000	26000
Aggregated		11000	12000	14000	18000	24000	28000	29000	30000

Statistika odziva sistema tokom promene režima rada veš mašine



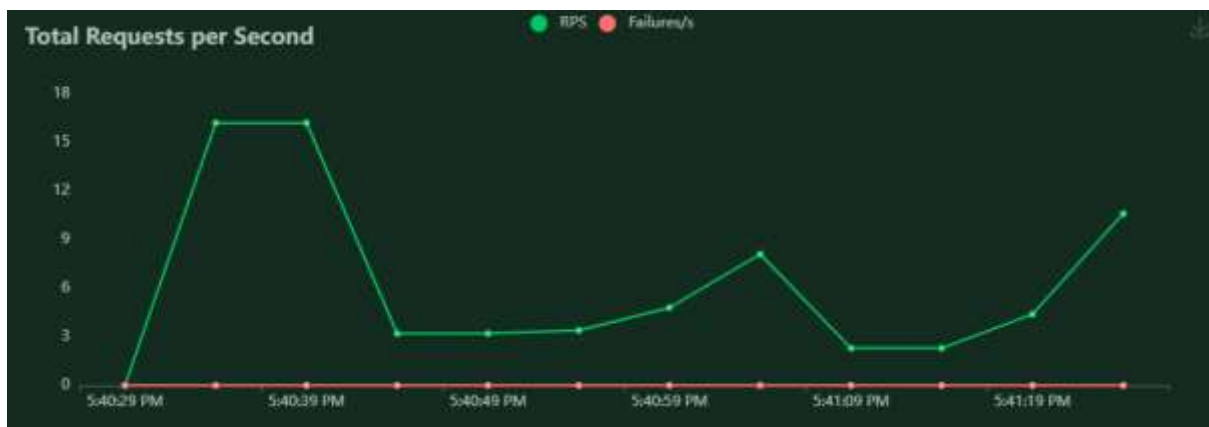
Grafik zahteva i padova u sekundi tokom vremena tokom promene režima rada veš mašine



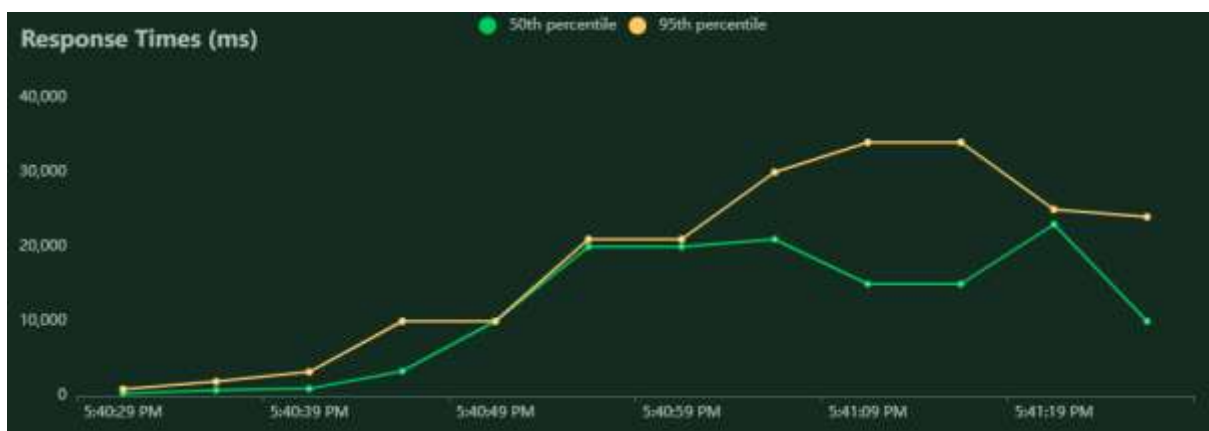
Grafik vremena odziva sistema tokom vremena tokom promene režima rada već mašine

Promena temperature rada klima uređaja

Testiranje promene temperature rada klima uređaja je neophodna operacija za moduliranje rada klima uređaja u pametnoj kući. Test se sastoji od prijavljivanja korisnika, nakon čega se šalje PUT zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju klima uređaja i novoj temperaturi rada klima uređaja. Najveći izazov ovog zahteva predstavlja izmena podataka u PostgreSQL bazi podataka i Influx bazi podataka. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.



Grafik zahteva i padova u sekundi tokom promene temperature rada klima uređaja



Grafik vremena odziva sistema tokom promene temperature rada klima uređaja

Dodavanje novog administratora

Testiranje dodavanja novog administratora je funkcionalnost neophodna za administraciju celokupnog sistema. Test sa sastoji od jednostavnog HTTP POST zahteva koji će slati razne podatke o novom adminu korisniku kao što su ime i prezime, kredencijali za prijavu, email adresa i fajl koji predstavlja profilnu sliku administratora. Nakon testiranja sistema dobili smo uspešnu identifikaciju u 100% slučajeva. Najveće zagušenje sistema je predstavljala obrada velikih slika i zapisivanje istih u perzistentno skladište. Jedan od načina za unapređenje ove funkcionalnosti je ograničavanje/kompersija profilne slike korisnika čime bi obrada velikih slika bila mnogo brža.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/User/addAdmin	525	0	6625	511	13792	382	9.0	0.0
POST	/api/User/login	100	0	4247	381	18352	23	1.7	0.0
	Aggregated	625	0	6412	381	18352	324	10.7	0.0

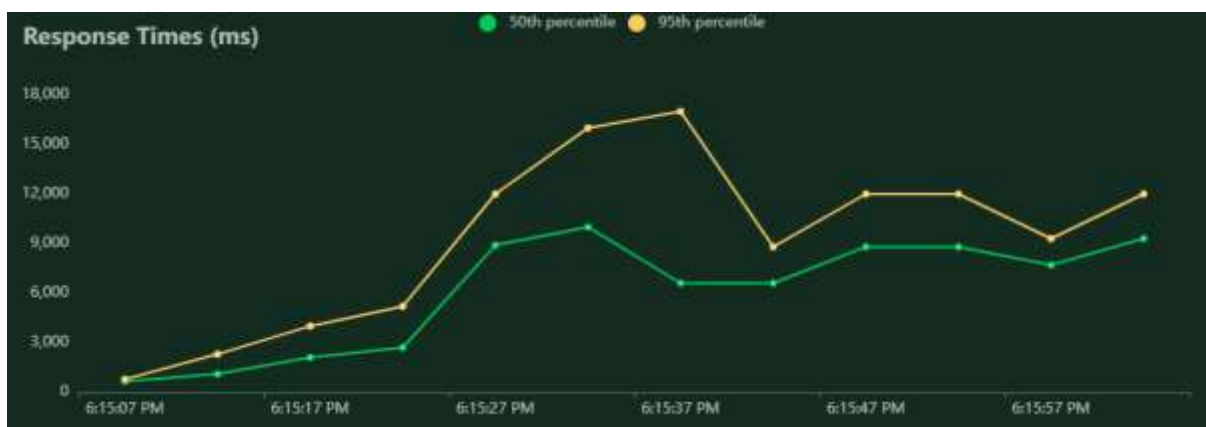
Statistika zahteva tokom dodavanja novog administratora

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/User/addAdmin	7500	8300	9200	10000	11000	12000	13000	14000
POST	/api/User/login	2100	2600	4100	5400	15000	17000	18000	18000
	Aggregated	6700	7900	8900	10000	11000	12000	16000	18000

Statistika odziva sistema tokom dodavanja novog administratora



Grafik zahteva i padova u sekundi tokom vremena tokom dodavanja novog administratora



Grafik vremena odziva sistema tokom vremena tokom dodavanja novog administratora

Promena šifre korisnika

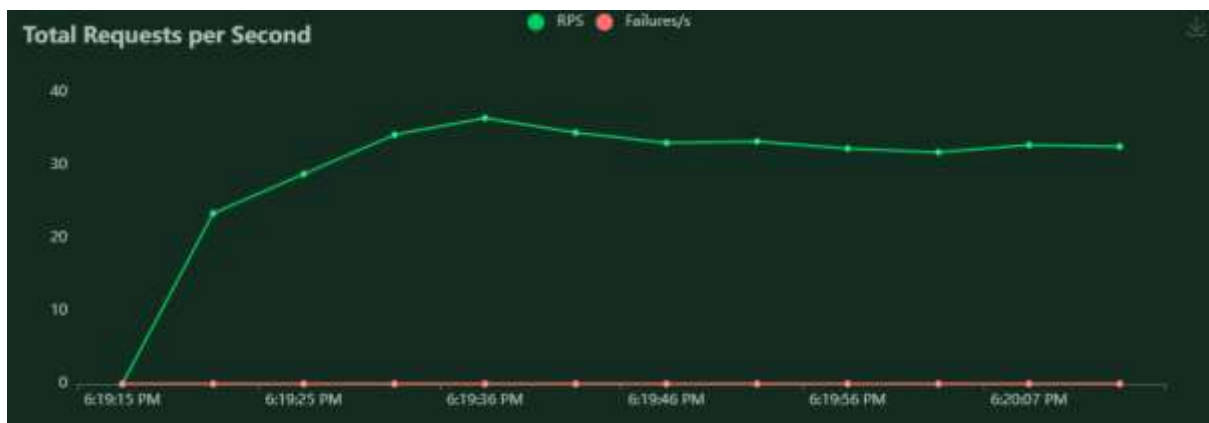
Testiranje promene šifre korisnika je kritična funkcionalnost neophodna za privatnost. Test sa sastoji od jednostavnog HTTP PUT zahteva koji će podatke o staroj i novoj šifri. Nakon testiranja sistema dobili smo uspešnost poziva u 100% slučajeva. Najveće zagušenje sistema je više hešovanja i dodavanja salt-ova. Jedan od načina za unapređenje ove funkcionalnosti hešovanje na klijentskoj strani.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/User/changePassword	1861	0	904	208	2634	30	30.9	0.0
POST	/api/User/login	100	0	719	323	2355	23	1.7	0.0
	Aggregated	1961	0	894	208	2634	29	32.6	0.0

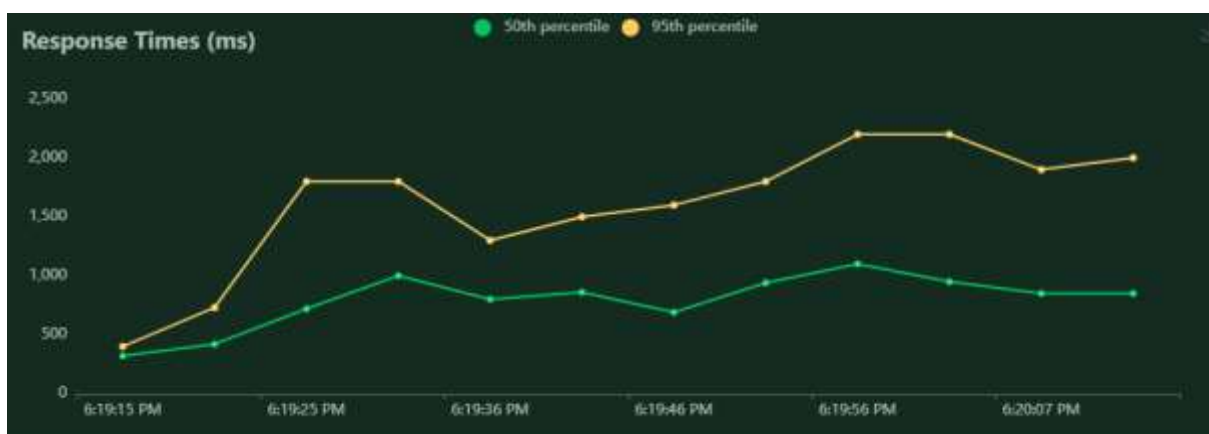
Statistika zahteva tokom promene šifre korisnika

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/User/changePassword	800	1000	1100	1300	1700	1900	2200	2600
POST	/api/User/login	510	700	1000	1100	1200	1700	2400	2400
	Aggregated	790	990	1100	1300	1700	1900	2200	2600

Statistika odziva sistema tokom promene šifre korisnika



Grafik zahteva i padova u sekundi tokom promene šifre korisnika



Grafik vremena odziva sistema tokom promene šifre korisnika

Student 2

Registracija pametne nekretnine

Testiranje registracije pametne nekretnine je operacija koja je esencijalna za ovakav sistem. Test se sastoji od prijavljivanja korisnika, nakon čega se šalje POST zahtev sa serveru pri čemu mu se šalju podaci o imenu nekretnine, adresi, državi i gradu u kome se nekretnina nalazi, površini i tipu nekretnine, broju spratova kao i geografskoj lokaciji nekretnine, ali i samoj fotografiji nekretnine. Najveće zagušenje sistema je predstavljala obrada velikih slika i zapisivanje istih u perzistentno skladište. Jedan od načina za unapređenje ove funkcionalnosti je ograničavanje/kompersija profilne slike korisnika čime bi obrada velikih slika bila mnogo brža. Nakon testiranja sistema dobili smo uspešnu identifikaciju u 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/SmartHome/CreateSmartHome	942	0	2578	143	7352	540	15.5	0.0
POST	/api/User/login	100	0	3955	1039	11563	23	1.6	0.0
Aggregated		1042	0	2710	143	11563	490	17.1	0.0

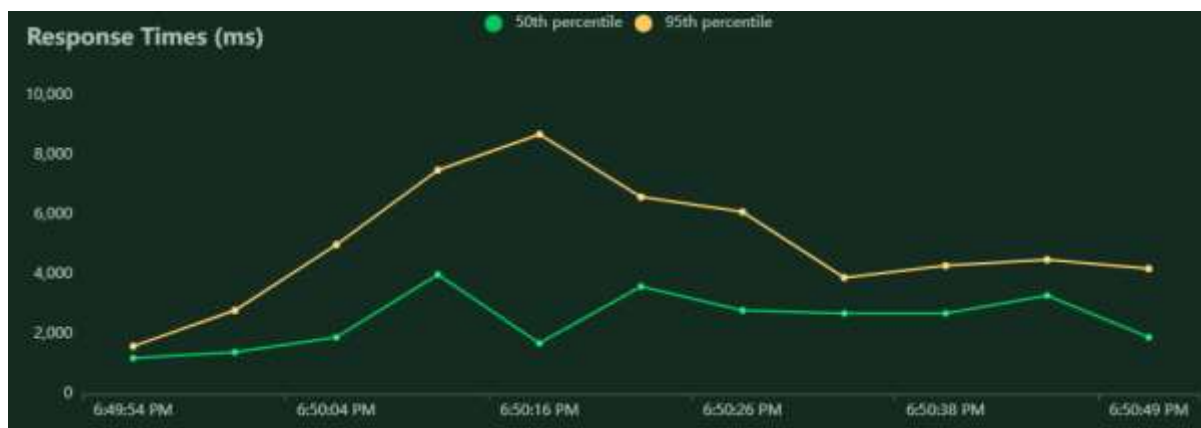
Statistika zahteva tokom registracije pametne nekretnine

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/SmartHome/CreateSmartHome	2300	2700	3100	3600	4300	5400	6600	7400
POST	/api/User/login	2500	3400	5500	7400	8500	9500	12000	12000
Aggregated		2300	2700	3200	3900	4700	6100	8300	12000

Statistika odziva sistema tokom registracije pametne nekretnine



Grafik zahteva i padova u sekundi tokom registracije pametne nekretnine



Grafik vremena odziva sistema tokom vremena tokom registracije pametne nekretnine

Provera dostupnosti uređaja

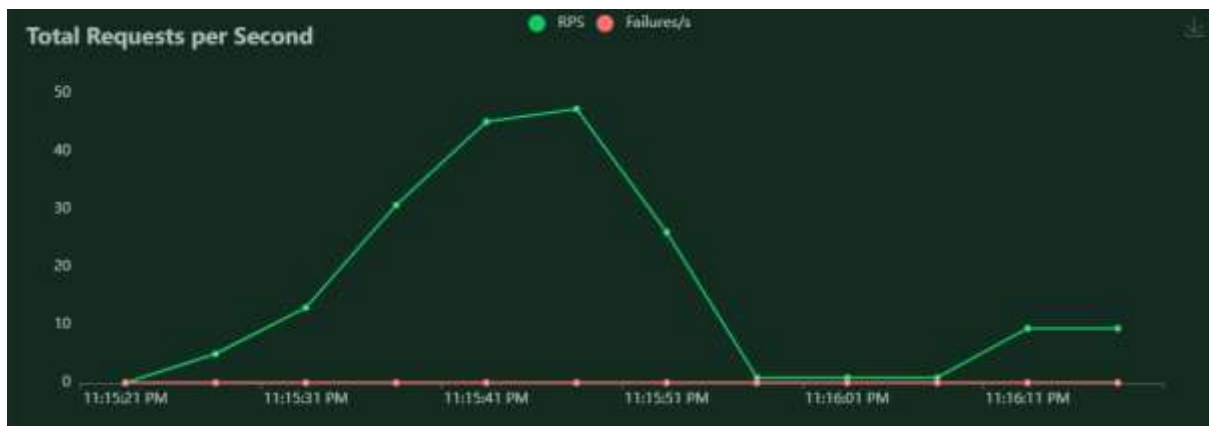
Testiranje provere dostupnosti uređaja je funkcionalnost značajna za određivanja kvaliteta komunikacije uređaja i servera kao i ključni pokazatelj neispravnosti uređaja. Test se sastoji od prijavljivanja korisnika nakon čega se šalje GET zahtev ka serveru pri čemu se šalju podaci o identifikacionom broju uređaja i vremenski period za koje je potrebno dobiti podatke o dostupnosti. Najveće opterećenje sistema predstavlja kompleksan upit ka Influx bazi podataka i potencijalna ogromna količina podataka koja se može dobiti kao povratna vrednost. Ovaj problem smo rešili maksimalnim iskorišćavanjem mogućnosti Influx upita kako bismo rasteretili server. Nakon testiranja sistema dobili smo uspešnu identifikaciju u 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/SmartDevice/GetAvailabilityData?id=ed6823db-34d6-43ba-ad6f-bc8a91ae874f&h=6h	629	0	740	53	2926	593	12.7	0.0
POST	/api/User/login	285	3	18817	1721	37239	22	5.8	0.1
Aggregated		914	3	6377	53	37239	415	18.5	0.1

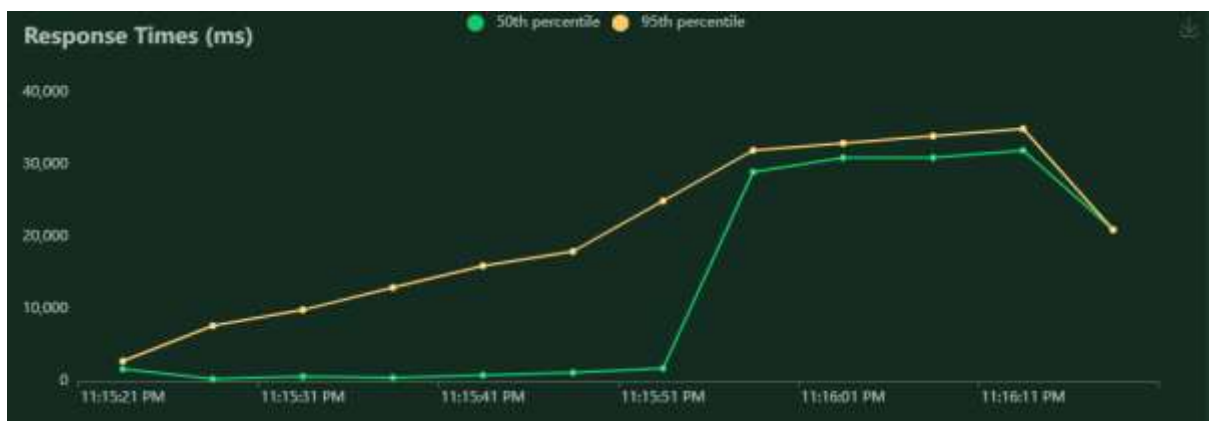
Statistika zahteva tokom provere dostupnosti uređaja

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	/api/SmartDevice/GetAvailabilityData?id=ed6823db-34d6-43ba-ad6f-bc8a91ae874f&h=6h	660	760	940	1100	1500	1800	2400	2900
POST	/api/User/login	17000	20000	29000	30000	32000	33000	35000	37000
Aggregated		1000	1400	4400	13000	28000	31000	33000	37000

Statistika odziva sistema tokom provere dostupnosti uređaja



Grafik zahteva i padova u sekundi tokom vremena tokom provere dostupnosti uređaja



Grafik vremena odziva sistema tokom vremena tokom provere dostupnosti uređaja

Dobavljanje podataka o istoriji akcija kapije

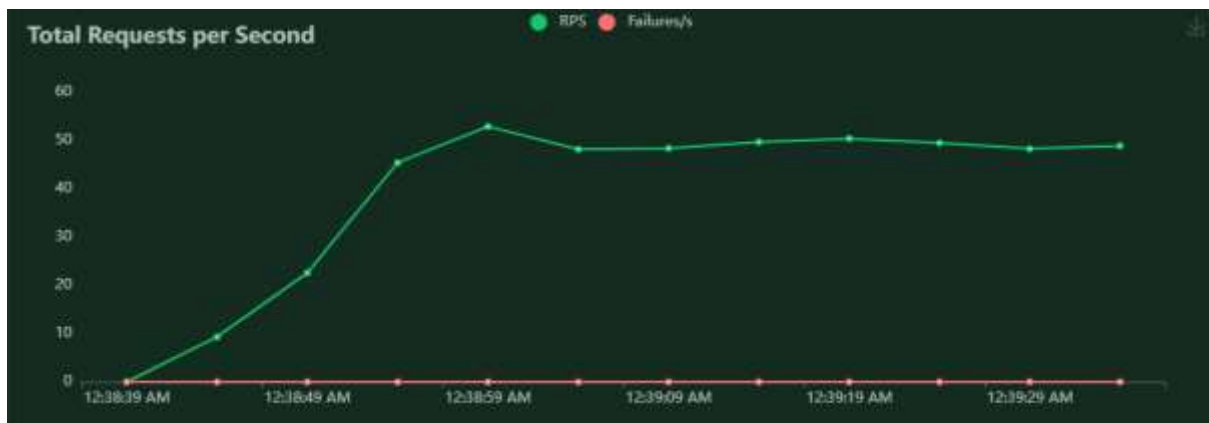
Testiranje dobavljanja podataka o istoriji akcija kapije je ključna operacija vezana za sigurnost pametnog doma. Test se sastoji od prijavljivanja korisnika, nakon čega se šalje GET zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju kapije kao i vremenski opseg za koje je potrebno dobiti podatke. Najveći izazov ovog zahteva za server predstavlja potencijalno dobavljanje velike količine podataka iz time-series Influx baze podataka. Ovaj problem je rešen uvođenjem određenog opsega iz kog se dobavljaju podaci, kao i ograničavanjem maksimalnog opsega na 30 dana. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/User/login	100	0	5362	1553	9365	23	1.7	0.0
GET	/api/VehicleGate/GetHistoricalActionData?id=ce456c90-3bb2-42be-bd2a-4d75a6990291&from=2023-01-01%2000:00:00&to=2023-12-31%2000:00:00	2615	0	49	16	776	3338	43.5	0.0
Aggregated		2715	0	245	16	9365	3215	45.2	0.0

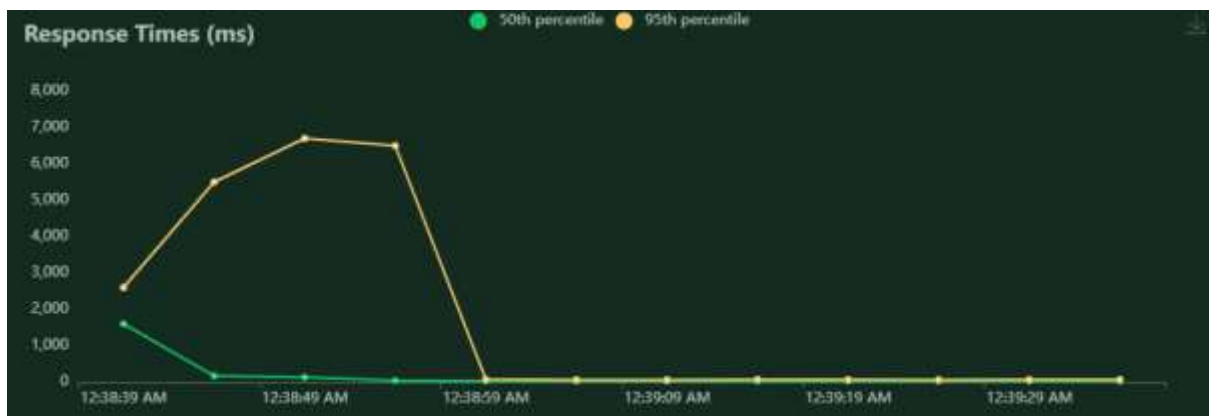
Statistika zahteva tokom dobavljanja istorije akcija kapije

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/User/login	5800	6000	6300	6600	7100	7600	9400	9400
GET	/api/VehicleGate/GetHistoricalActionData?id=ce456c90-3bb2-42be-bd2a-4d75a6990291&from=2023-01-01%2000:00:00&to=2023-12-31%2000:00:00	31	33	38	51	69	130	460	780
Aggregated		31	34	41	55	93	380	6400	9400

Statistika odziva sistema tokom *dobavljanja istorije akcija kapije*



Grafik zahteva i padova u sekundi tokom vremena tokom *dobavljanja istorije akcija kapije*



Grafik vremena odziva sistema tokom vremena tokom *dobavljanja istorije akcija kapije*

Dobavljanje podataka o lampi

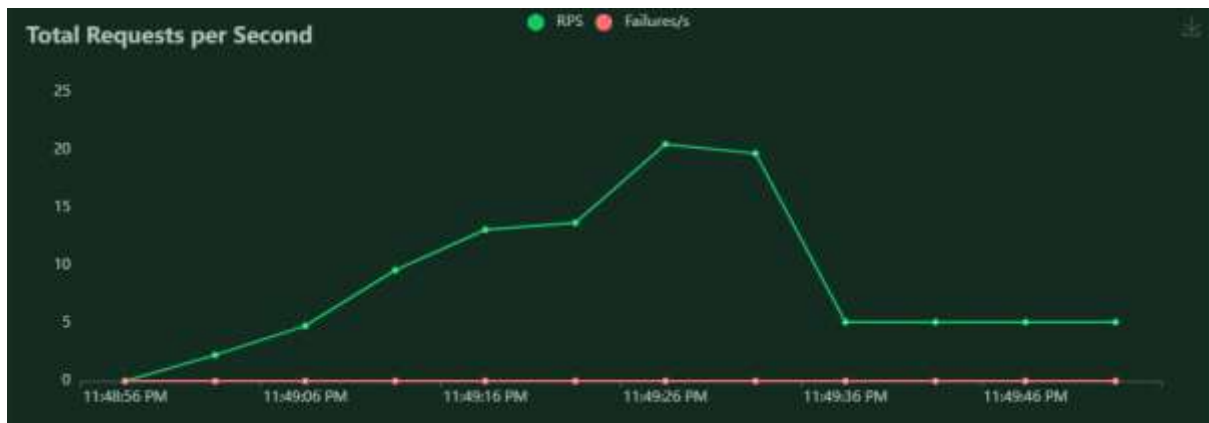
Testiranje dobavljanja podataka o lampi je osnovna operacija u radu sa lampom u pametnoj kući. Test se sastoji od prijavljivanja korisnika, nakon čega se šalje GET zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju lampe. Najveći izazov ovog zahteva za server predstavlja dobavljanje podataka iz dva različita perzistentna skladišta. Potrebno je dobiti podatke o automatskom režimu rada lampe iz PostgreSQL baze, kao i trenutne podatke o osvetljenosti i radu lampe iz time-series Influx baze podataka. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RP5	Failures/s
GET	/api/Lamp/Get?id=ac7ab1e0-d9f7-4cd2-b297-30059e10b35c	146	0	10680	47	14232	283	3.6	0.0
POST	/api/User/login	312	1	16117	2371	32999	35	7.8	0.0
Aggregated		458	1	14384	47	32999	114	11.4	0.0

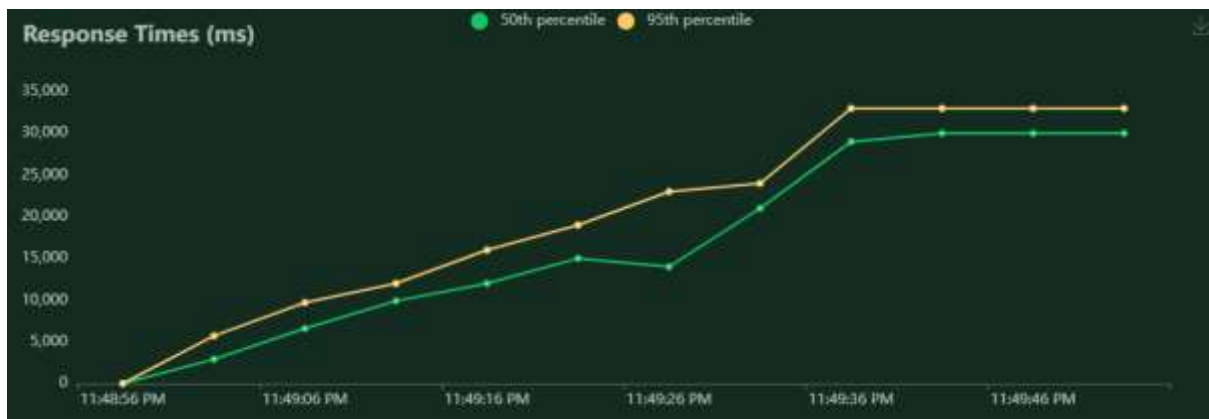
Statistika zahteva tokom dobavljanja podataka o lampi

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	/api/Lamp/Get?id=ac7ab1e0-d9f7-4cd2-b297-30059e10b35c	12000	13000	13000	14000	14000	14000	14000	14000
POST	/api/User/login	15000	17000	20000	22000	28000	30000	32000	33000
Aggregated		13000	14000	16000	20000	24000	29000	32000	33000

Statistika odziva sistema tokom dobavljanja podataka o lampi



Grafik zahteva i padova u sekundi tokom vremena tokom dobavljanja podataka o lampi



Grafik vremena odziva sistema tokom vremena tokom dobavljanja podataka o lampi

Otvaranje/Zatvaranje kapije

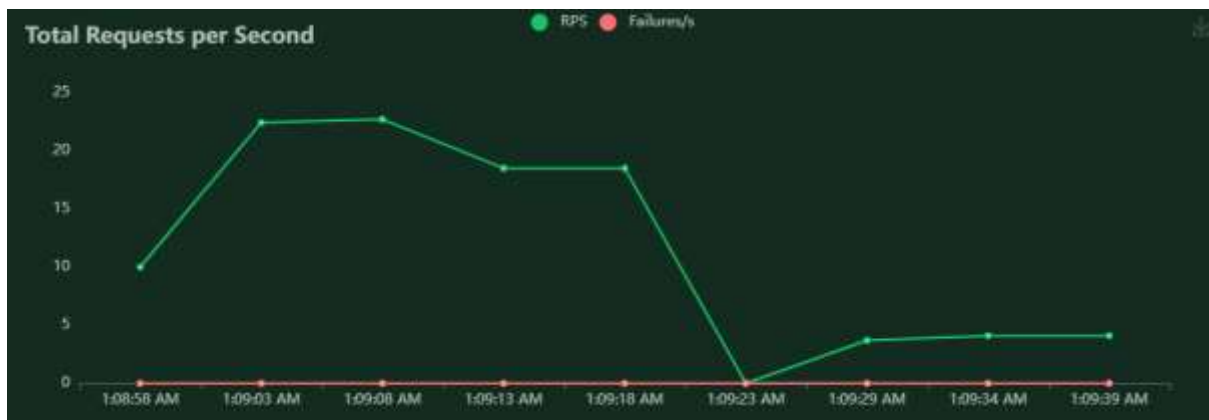
Test se sastoji od prijavljivanja korisnika, nakon čega se šalje PUT zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju kapije i binarna vrednost na koje stanje treba da se promeni stanje kapije. Najveći izazov ovog zahteva za server predstavlja dobavljanje i promena podataka iz dva različita perzistentna skladišta. Potrebno je dobiti podatke o postojanju takvog uređaja, zatim dodati ažuriranu vrednost u time-series bazu podataka i tu informaciju objaviti. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/User/login	99	0	3068	945	17982	23	2.8	0.0
PUT	/api/VehicleGate/Toggle?id=ce456c90-3bb2-42be-bd2a-4d75a6990291&turnOn=True	216	1	4212	49	25415	16	6.0	0.0
Aggregated		315	1	3852	49	25415	18	8.8	0.0

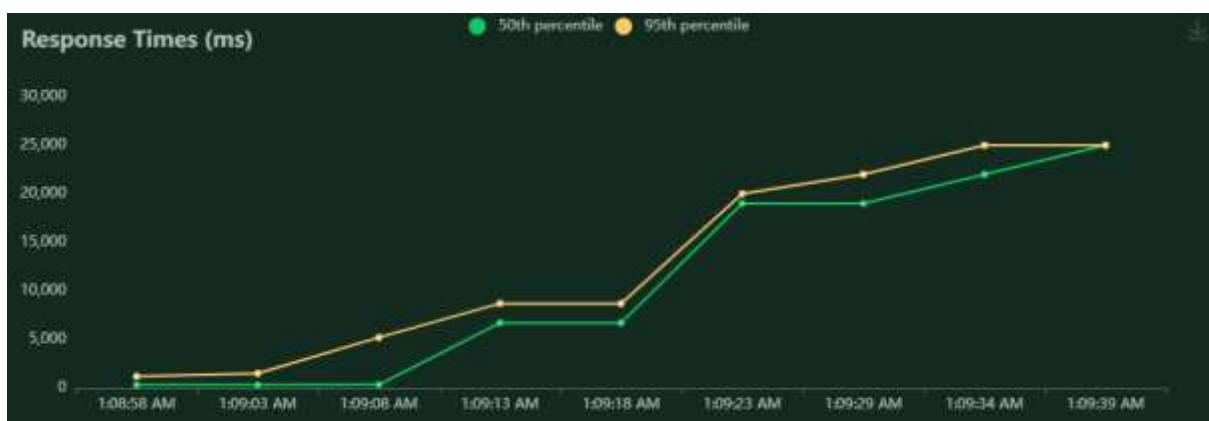
Statistika zahteva tokom otvaranja/zatvaranja kapije

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/User/login	1400	1500	2400	5600	7700	8700	18000	18000
PUT	/api/VehicleGate/Toggle?id=ce456c90-3bb2-42be-bd2a-4d75a6990291&turnOn=True	250	320	440	6900	19000	21000	25000	25000
Aggregated		470	1100	1400	5600	19000	20000	25000	25000

Statistika odziva sistema tokom otvaranja/zatvaranja kapije



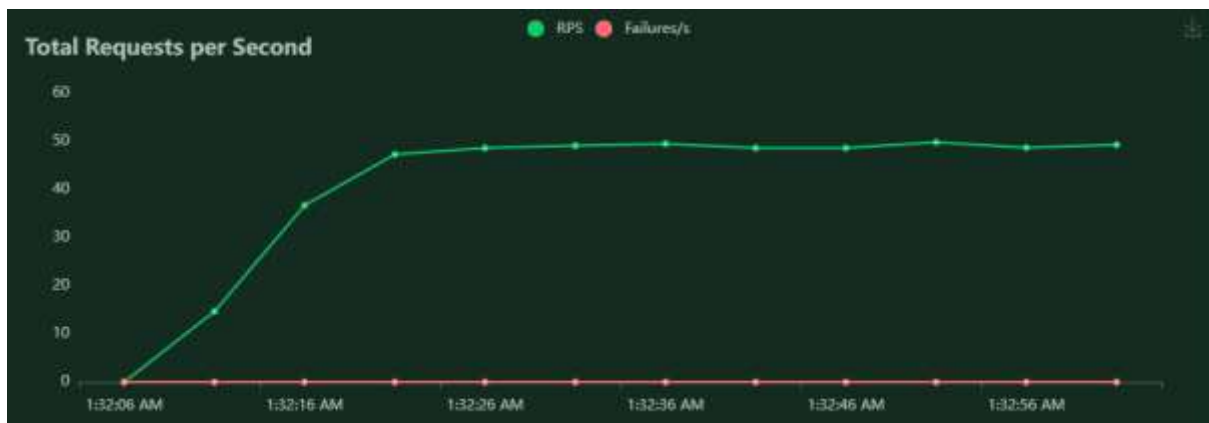
Grafik zahteva i padova u sekundi tokom vremena tokom otvaranja/zatvaranja kapije



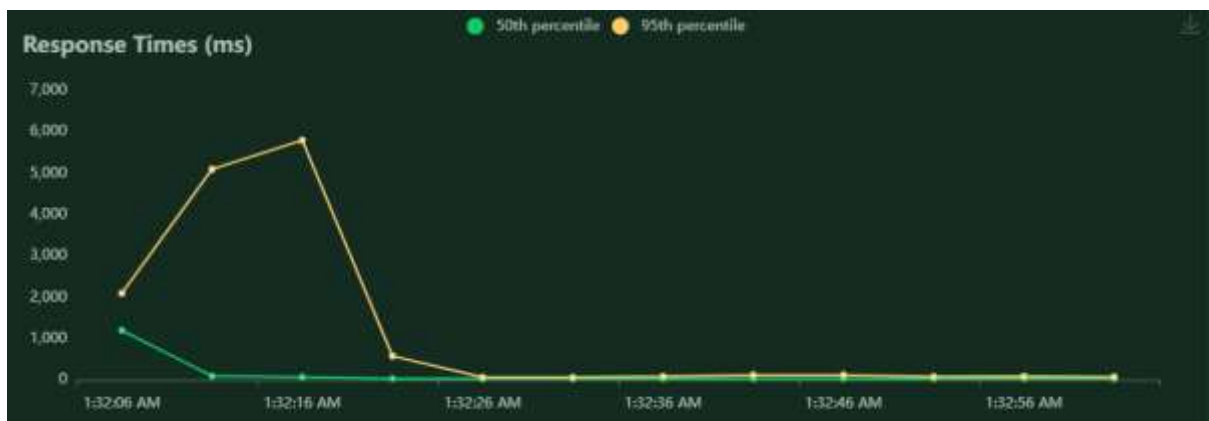
Grafik vremena odziva sistema tokom vremena tokom otvaranja/zatvaranja kapije

Dodavanje verifikovane tablice za kapiju

Test se sastoji od jednostavnog HTTP PUT zahteva koji će slati podatke o novoj, verifikovanoj tablici, uz identifikacioni broj kapije. Najveći izazov ove funkcionalnosti je izmena i objavljivanje novih akcija što opterećuje mqtt komunikaciju kao i sam server. Nakon testiranja sistema dobili smo uspešnu identifikaciju u 100% slučajeva.



Grafik zahteva i padova u sekundi tokom dodavanja verifikovane tablice



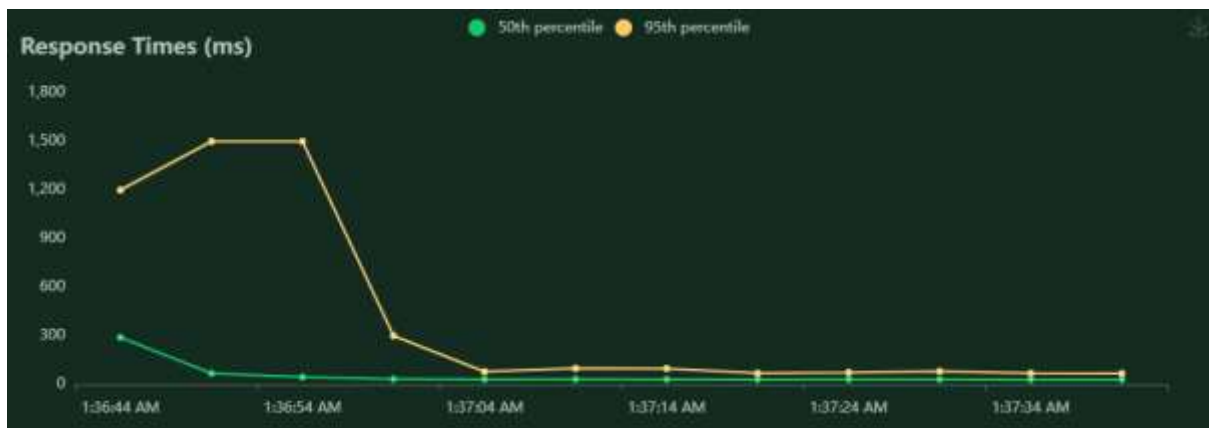
Grafik vremena odziva sistema tokom vremena tokom dodavanja verifikovane tablice

Brisanje verifikovane tablice za kapiju

Test se sastoji od jednostavnog HTTP PUT zahteva koji će slati podatke o tablici koju je potrebno obrisati, verifikovanoj tablici, uz identifikacioni broj kapije. Najveći izazov ove funkcionalnosti je izmena i objavljivanje novih akcija što opterećuje mqtt komunikaciju kao i sam server. Nakon testiranja sistema dobili smo uspešnu identifikaciju u 100% slučajeva.



Grafik zahteva i padova u sekundi tokom brisanja verifikovane tablice



Grafik vremena odziva sistema tokom vremena tokom brisanja verifikovane tablice

Dobavljanje svih nekretnina jednog korisnika

Test se sastoji od prijavljivanja korisnika, nakon čega se šalje GET zahtev ka serveru koji iz cookie-a vadi podatke o korisniku. Napredak koji je značajno ubrzao rad ovog test slučaja je paginacija filtriranih uređaja na backend-u čime se u velikom broju smanjila količina podataka koja se šalje preko mreže i keširanje sadržaja pomoću redis-a. Nakon testiranja sistema dobili smo uspešnost od 99% slučajeva.

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	/api/SmartHome/GetSmartHomesForUser?PageNumber=1&PageSize=10&search=%22%22	39	48	65	95	140	170	280	18000
POST	/api/User/login	720	740	790	870	1000	1200	1300	1300
Aggregated		45	60	88	140	580	730	1000	18000

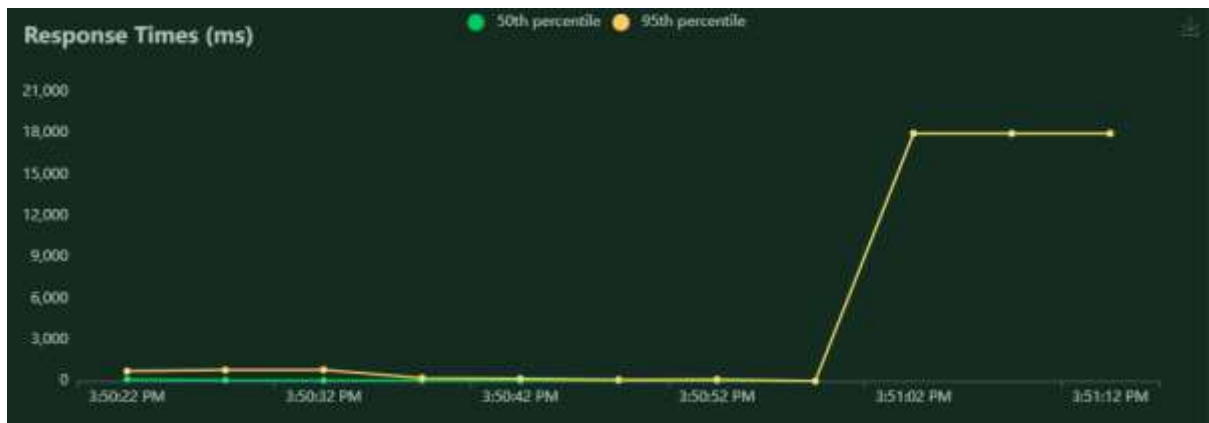
Statistika zahteva tokom dobavljanja svih nekretnina jednog korisnika

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/SmartHome/GetSmartHomesForUser?PageNumber=1&PageSize=10&search=%22%22	825	1	83	14	18064	1106	20.6	0.0
POST	/api/User/login	100	0	759	541	1340	23	2.5	0.0
Aggregated		925	1	156	14	18064	989	23.1	0.0

Statistika odziva sistema tokom dobavljanja svih nekretnina jednog korisnika



Grafik zahteva i padova u sekundi tokom dobavljanja svih nekretnina jednog korisnika



Grafik vremena odziva sistema tokom vremena tokom dobavljanja svih nekretnina jednog korisnika

Kreiranje zakazanih termina za rad prskalice

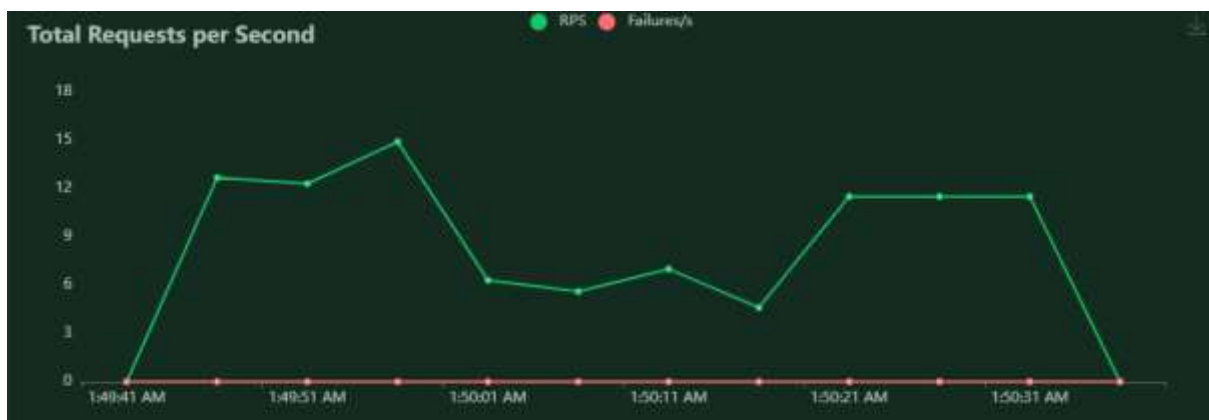
Test se sastoji od jednostavnog HTTP PUT zahteva koji će slati podatke o terminu kada je neophodno da se prskalice pali, koliko dugo da radi. Najveći izazov ove funkcionalnosti je izmena i objavljivanje novih akcija što opterećuje mqtt komunikaciju kao i sam server. Nakon testiranja sistema dobili smo uspešnu identifikaciju u 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/Sprinkler/AddScheduledWork	349	0	10029	72	20709	0	5.9	0.0
POST	/api/User/login	100	0	5610	1402	9411	23	1.7	0.0
Aggregated		449	0	9045	72	20709	5	7.6	0.0

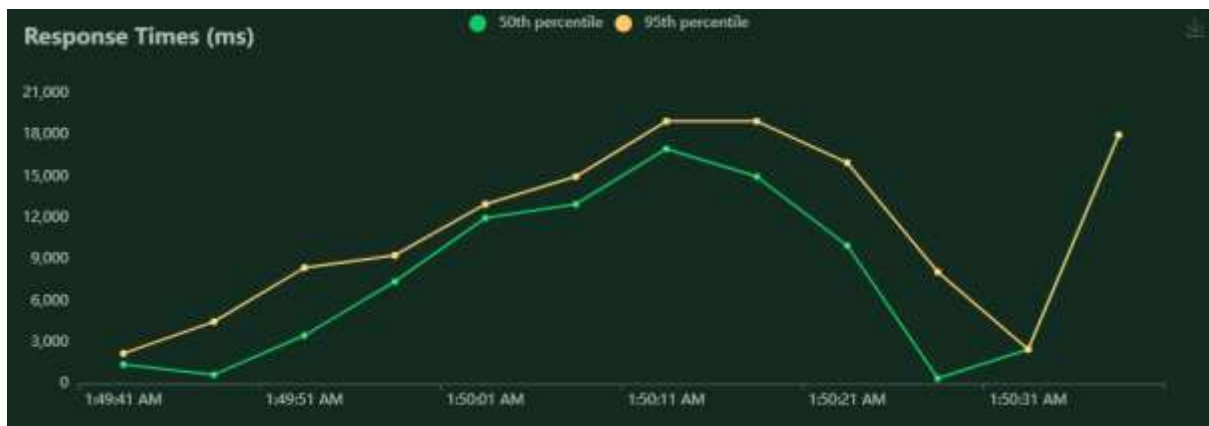
Statistika zahteva tokom kreiranja zakazanih termina za rad prskalice

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/Sprinkler/AddScheduledWork	12000	13000	15000	17000	18000	18000	19000	21000
POST	/api/User/login	6400	7200	7400	8300	8400	8400	9400	9400
Aggregated		8400	12000	14000	16000	18000	18000	19000	21000

Statistika odziva sistema tokom kreiranja zakazanih termina za rad prskalice



Grafik zahteva i padova u sekundi tokom kreiranja zakazanih termina za rad prskalice



Grafik vremena odziva sistema tokom vremena tokom kreiranja zakazanih termina za rad prskalice

Promena režima rada lampe

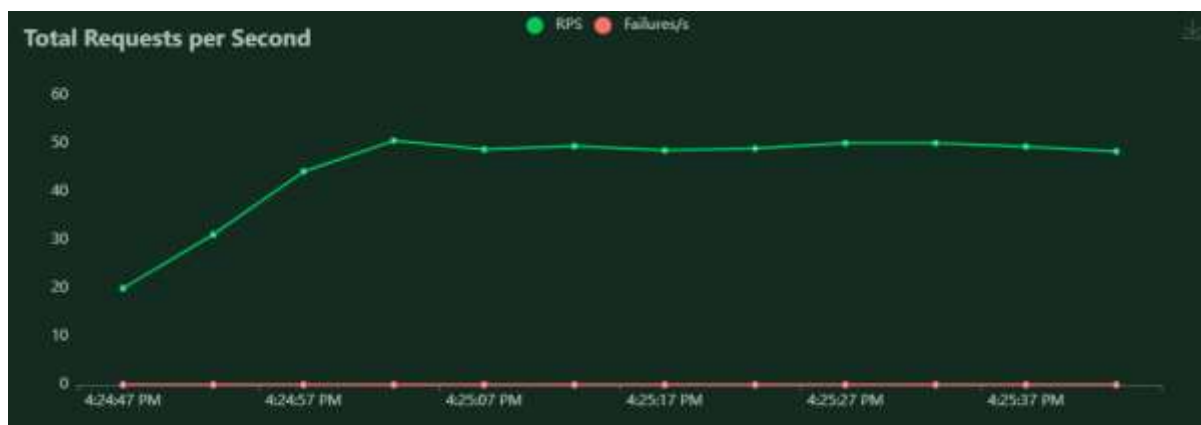
Testiranje promene režima rada lampe je neophodna operacija za moduliranje rada lampe u pametnoj kući. Test se sastoji od prijavljivanja korisnika, nakon čega se šalje PUT zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju lampe i novom režimu rada lampe. Najveći izazov ovog zahteva predstavlja izmena podataka u PostgreSQL bazi podataka i Influx bazi podataka. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
PUT	/api/Lamp/ChangeMode?id=027bfe12-7eb4-42e8-88e2-9dd0025464f8&isAuto=false	1355	0	19	7	282	0	22.6	0.0
PUT	/api/Lamp/ChangeMode?id=027bfe12-7eb4-42e8-88e2-9dd0025464f8&isAuto=true	1391	0	20	7	384	0	23.2	0.0
POST	/api/User/login	100	0	666	522	983	23	1.7	0.0
Aggregated		2846	0	42	7	983	0	47.4	0.0

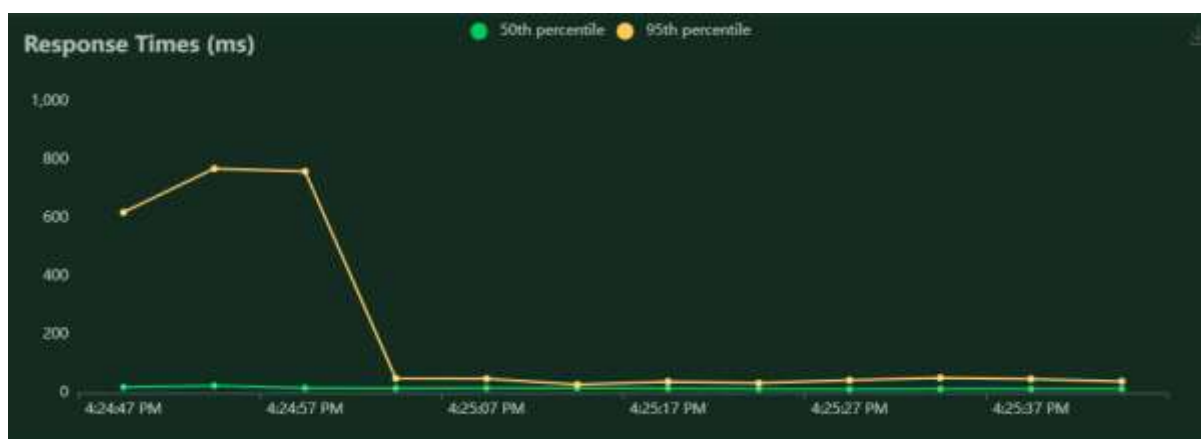
Statistika zahteva tokom promena režima rada lampe

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	98%ile (ms)	100%ile (ms)
PUT	/api/Lamp/ChangeMode?id=027bfe12-7eb4-42e8-88e2-9dd0025464f8&isAuto=false	15	17	20	24	33	43	74	280
PUT	/api/Lamp/ChangeMode?id=027bfe12-7eb4-42e8-88e2-9dd0025464f8&isAuto=true	15	17	19	23	33	47	120	380
POST	/api/User/login	620	690	730	770	810	860	980	980
Aggregated		16	17	20	25	39	63	740	980

Statistika odziva sistema tokom promena režima rada lampe



Grafik zahteva i padova u sekundi tokom promena režima rada lampe



Grafik vremena odziva sistema tokom vremena tokom promena režima rada lampe

Student 3

Dobavljanje podataka o bateriji u pametnoj kući

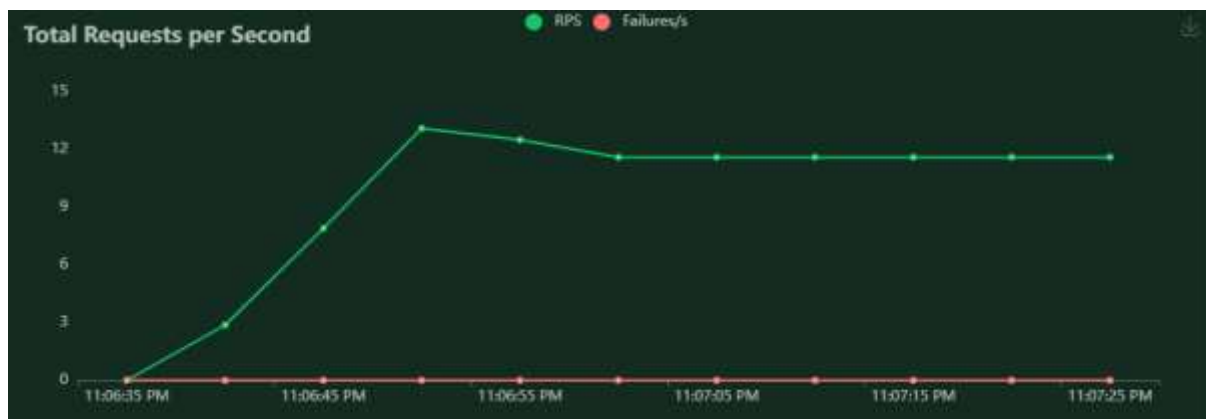
Testiranje dobavljanja podataka o bateriji je osnovna operacija u radu sa baterijom u pametnoj kući. Test se sastoji od prijavljivanja korisnika, nakon čega se šalje GET zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju baterije. Najveći izazov ovog zahteva za server predstavlja dobavljanje podataka iz dva različita perzistentna skladišta. Potrebno je dobiti podatke o maksimalnom kapacitetu baterije iz PostgreSQL baze, kao i trenutne podatke o kapacitetu baterije iz time-series Influx baze podataka. Nakon testiranja sistema dobili smo uspešnost od 99% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/BatterySystem/Get?id=ed6823db-34d6-43ba-ad6f-bc8a91ae874f	47	0	8738	160	14638	240	1.6	0.0
POST	/api/User/login	234	2	13195	2316	22669	55	8.2	0.1
Aggregated		281	2	12450	160	22669	86	9.8	0.1

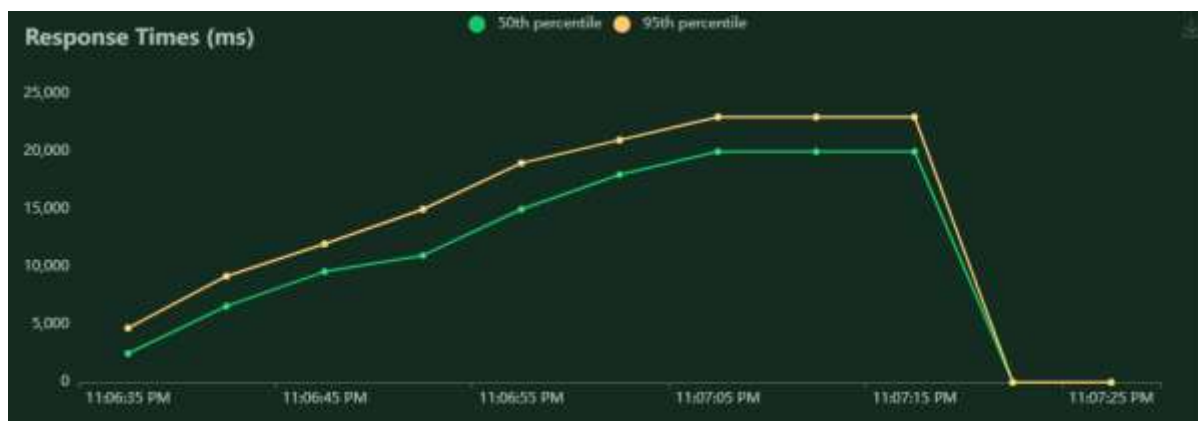
Statistika zahteva tokom dobavljanja podataka o bateriji

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	/api/BatterySystem/Get?id=ed6823db-34d6-43ba-ad6f-bc8a91ae874f	8600	8900	11000	14000	14000	15000	15000	15000
POST	/api/User/login	13000	15000	17000	18000	19000	20000	22000	23000
Aggregated		12000	14000	16000	17000	19000	20000	22000	23000

Statistika odziva sistema tokom dobavljanja podataka o bateriji



Grafik zahteva i padova u sekundi tokom vremena tokom dobavljanja podataka o bateriji



Grafik vremena odziva sistema tokom vremena tokom dobavljanja podataka o bateriji

Paljenje/Gašenje Solarnih Panela

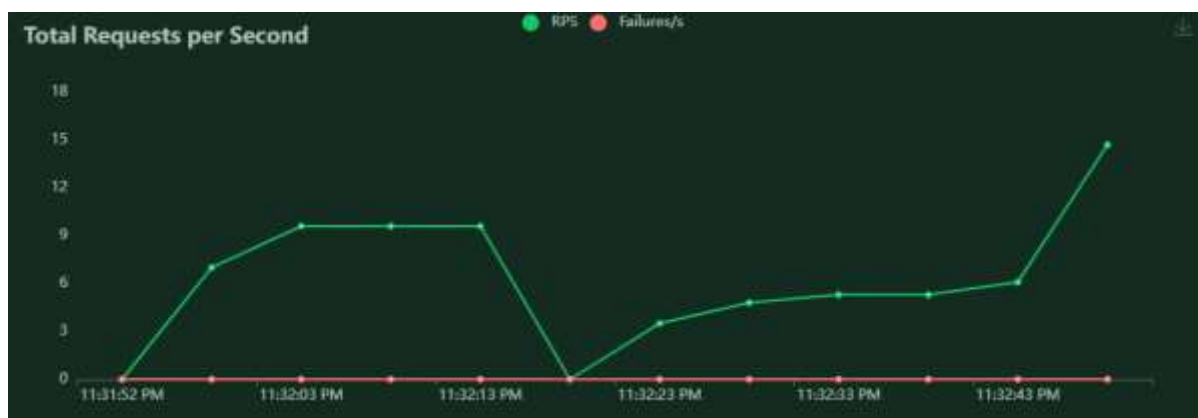
Test se sastoji od prijavljivanja korisnika, nakon čega se šalje PUT zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju solarnog panela i binarna vrednost na koje stanje treba da se promeni napajanje solarnog panela. Najveći izazov ovog zahteva za server predstavlja dobavljanje i promena podataka iz dva različita perzistentna skladišta. Potrebno je dobiti podatke o postojanju takvog uređaja, zatim dodati ažuriranu vrednost u time-series bazu podataka i tu informaciju javiti uređaju preko mqtt-a. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
PUT	/api/SolarPanelSystem/Toggle?id=83d1dd00-7f15-42d3-ab66-60802292d0ad&turnOn=True	320	0	12144	59	31198	0	5.7	0.0
POST	/api/User/login	100	0	8790	878	47235	23	1.8	0.0
Aggregated		420	0	11346	59	47235	5	7.4	0.0

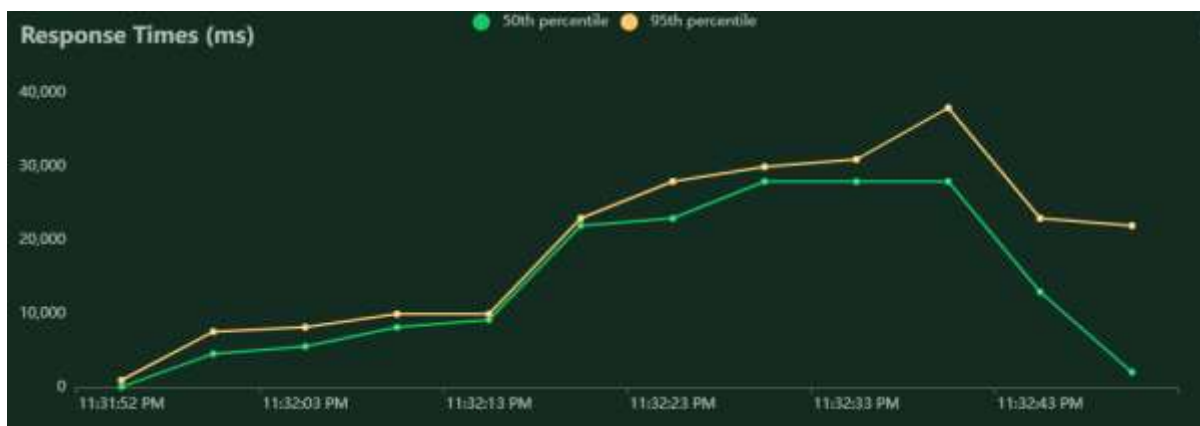
Statistika zahteva tokom promene stanja solarnog panela

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
PUT	/api/SolarPanelSystem/Toggle?id=83d1dd00-7f15-42d3-ab66-60802292d0ad&turnOn=True	13000	17000	21000	23000	28000	28000	29000	31000
POST	/api/User/login	5600	6600	7100	8200	28000	32000	47000	47000
Aggregated		7600	13000	20000	23000	28000	28000	32000	47000

Statistika odziva sistema tokom promene stanja solarnog panela



Grafik zahteva i padova u sekundi tokom vremena tokom promene stanja solarnog panela



Grafik vremena odziva sistema tokom vremena tokom promene stanja solarnog panela

Registracija uređaja

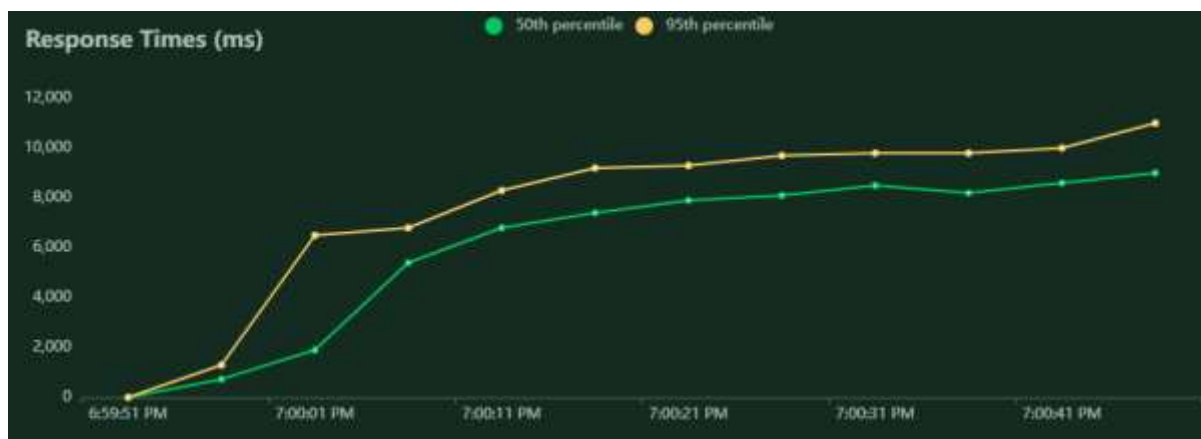
Test se sastoji od prijavljivanja korisnika, nakon čega se šalje POST zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju kuće u kojoj se registruje i osnovni podaci o uređaju sa akcetnom na sliku uređaja. Najveći izazov ovog zahteva za kreiranje uređaja je perzistiranje svih podataka, kreiranje kanala za komunikaciju pomoću web socketa, kao i kreiranje uređaja na samom simulatoru, potrebno je obratiti pažnju i na perzistiranje slike koja može oduzimati veliki procenat procesorskog vremena. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/SPUI/CreateVehicleGate/603223ce-8a82-41d6-aadc-9f96dfce35d9	494	0	7914	4436	12388	233	8.3	0.0
POST	/api/User/login	100	0	1860	623	5107	23	1.7	0.0
Aggregated		594	0	6895	623	12388	197	9.9	0.0

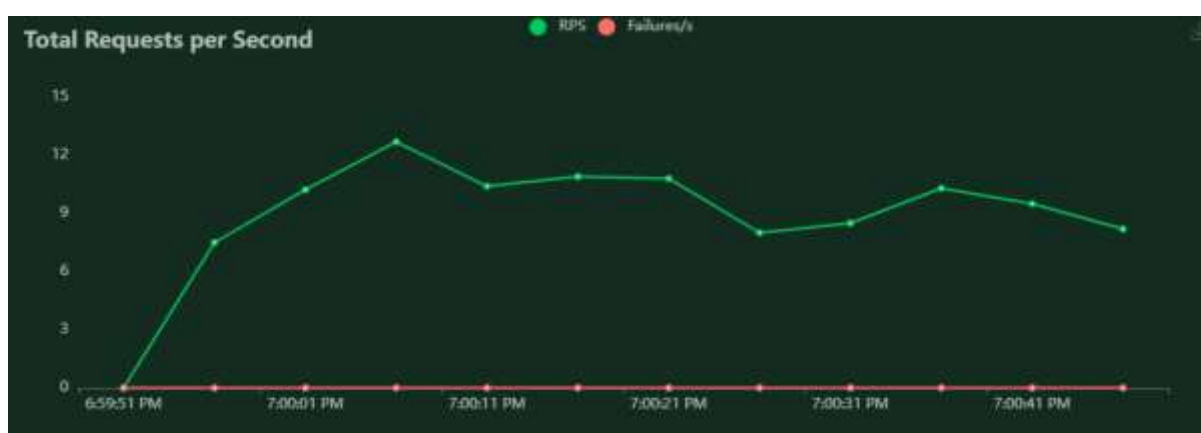
Statistika zahteva tokom registracije uređaja

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/SPUI/CreateVehicleGate/603223ce-8a82-41d6-aadc-9f96dfce35d9	7900	8300	8700	9000	9500	9900	11000	12000
POST	/api/User/login	1000	1700	2300	3700	4600	4600	5100	5100
Aggregated		7600	8000	8500	8500	9400	9800	11000	12000

Statistika odziva sistema tokom registracije uređaja



Grafik zahteva i padova u sekundi tokom vremena tokom registracije uređaja



Grafik vremena odziva sistema tokom vremena tokom registracije uređaja

Proizvodnja/Potrošnja svih uređaja unutar nekretnine

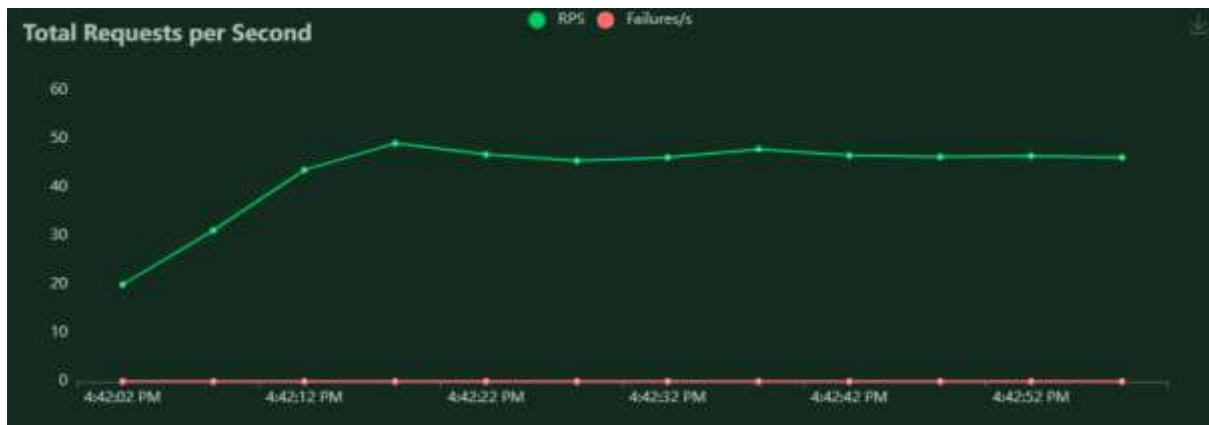
Testiranje dobavljanja podataka o proizvodnji/potrošnji svih uređaja unutar jedne nekretnine je neophodan za detaljan pregled informacija o toku električne energije. Test se sastoji od prijavljivanja korisnika, nakon čega se šalje GET zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju nekretnine za koju je potrebno dobiti podatke. Najveći izazov ovog zahteva predstavlja potencijalno dobavljanje velike količine podataka iz time-series Influx baze podataka sa više mesta, za potrošnju, ali i za potencijalnu proizvodnju ukoliko se u okviru nekretnine nalazi i solarni panel. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/City/GetCityHistoricalData?id=43ef2fe2-750b-4889-9f3f-12155a0e8e92&from=2023-01-01%2000:00:00&to=2023-12-31%2000:00:00	2619	0	126	28	784	2	43.7	0.0
POST	/api/User/login	100	0	547	296	1250	23	1.7	0.0
Aggregated		2719	0	142	28	1250	2	45.3	0.0

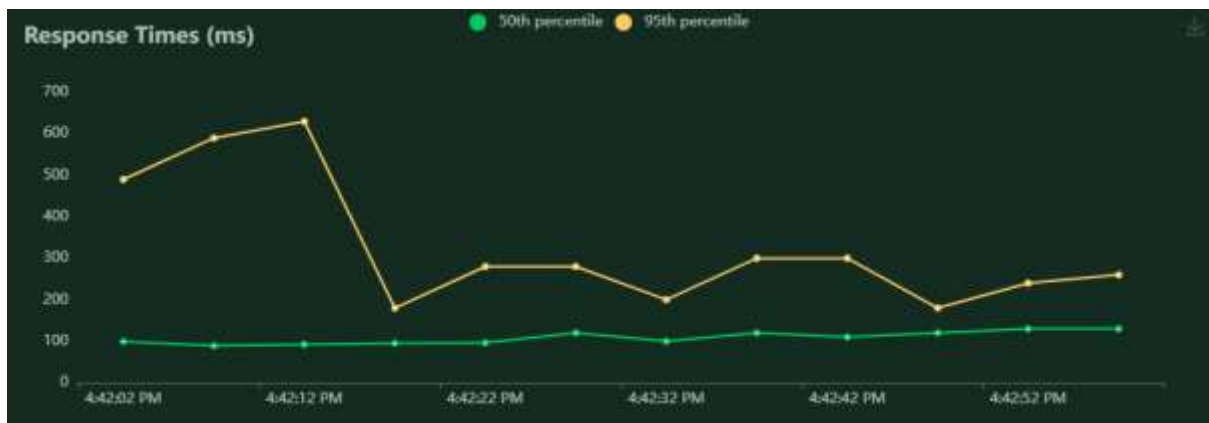
Statistika zahteva tokom dobavljanja potrošnje/proizvodnje svih uređaja unutar nekretnine

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	/api/City/GetCityHistoricalData?id=43ef2fe2-750b-4889-9f3f-12155a0e8e92&from=2023-01-01%2000:00:00&to=2023-12-31%2000:00:00	110	120	140	170	200	260	460	780
POST	/api/User/login	480	590	650	680	770	940	1300	1300
	Aggregated	110	130	150	180	230	370	670	1300

Statistika odziva sistema tokom dobavljanja potrošnje/proizvodnje svih uređaja unutar nekretnine



Grafik zahteva i padova u sekundi tokom vremena tokom dobavljanja potrošnje/proizvodnje svih uređaja unutar nekretnine



Grafik vremena odziva sistema tokom vremena tokom dobavljanja potrošnje/proizvodnje svih uređaja unutar nekretnine

Dodavanje automobila na punjač

Test se sastoji od jednostavnog HTTP PUT zahteva koji će slati podatke o automobili u njegovima karakteristikama, uz identifikacioni punjača. Nakon toga se obrađuju podaci o punjaču i dostupnosti njegovih utičnica. Najveći izazov ove funkcionalnosti je izmena podatka u oba perzistentna skladišta i javljanje samom uređaju o novim akcijama što opterećuje mqtt komunikaciju kao i sam server. Nakon testiranja sistema dobili smo uspešnu identifikaciju u 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/User/login	100	0	1298	624	2935	23	1.7	0.0
PUT	/api/VehicleCharger/ConnectToCharger?vehicleChargerId=310f5632-dc3a-4d8f-89ba-a6d0ba897478&vehicleChargingPointId=84ecfe09-956b-487e-8d1b-4d3060797c34	1350	0	2017	785	7233	51	22.5	0.0
Aggregated		1450	0	1968	624	7233	49	24.2	0.0

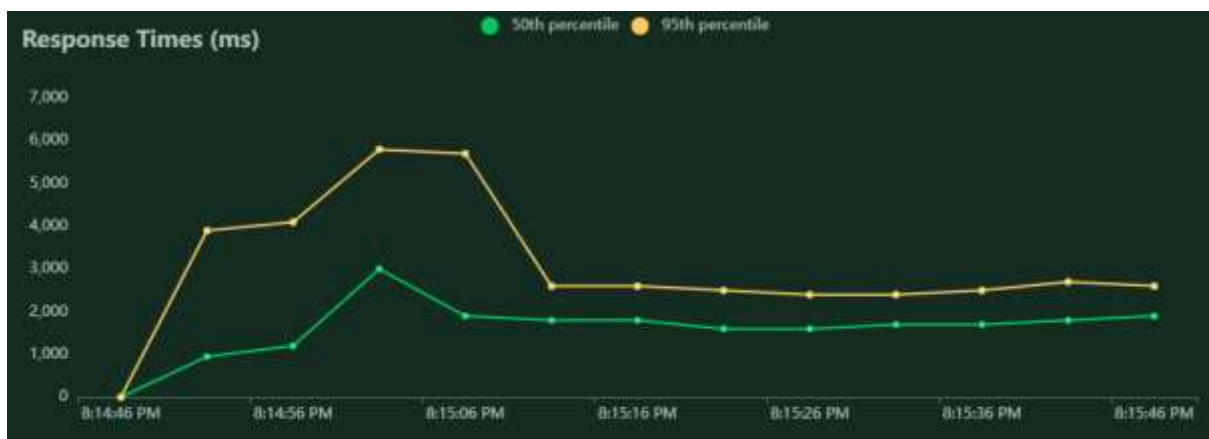
Statistika zahteva tokom dodavanja automobila na punjač

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/User/login	1200	1200	1300	1800	1900	2700	2900	2900
PUT	/api/VehicleCharger/ConnectToCharger?vehicleChargerId=310f5632-dc3a-4d8f-89ba-a6d0ba897478&vehicleChargingPointId=84ecfe09-956b-487e-8d1b-4d3060797c34	1800	1900	2000	2300	2900	4100	5700	7200
Aggregated		1800	1900	2000	2300	2900	4100	5700	7200

Statistika odziva sistema tokom dodavanja automobila na punjač



Grafik zahteva i padova u sekundi tokom vremena tokom dodavanja automobila na punjač



Grafik vremena odziva sistema tokom vremena tokom dodavanja automobila na punjač

Dobavljanje uređaja u okviru kuće

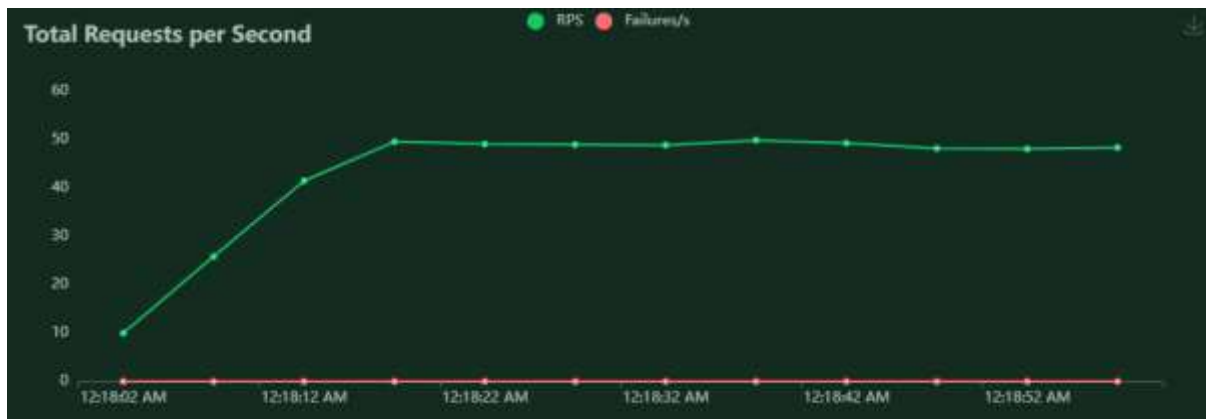
Test se sastoji od prijavljivanja korisnika, nakon čega se šalje GET zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju kuće. Napredak koji je značajno ubrzao rad ovog test slučaja je paginacija filtriranih uređaja na backend-u čime se u velikom broju smanjila količina podataka koja se šalje preko mreže i keširanje sadržaja pomoću redis-a. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/SmartDevice/GetSmartDevicesForHome/f901e6d3-9fba-4506-a23a-06c91ab97a277?PageNumber=1&PageSize=10	2810	0	51	15	846	2063	45.0	0.0
POST	/api/User/login	100	0	1363	989	2889	23	1.6	0.0
Aggregated		2910	0	96	15	2889	1992	46.6	0.0

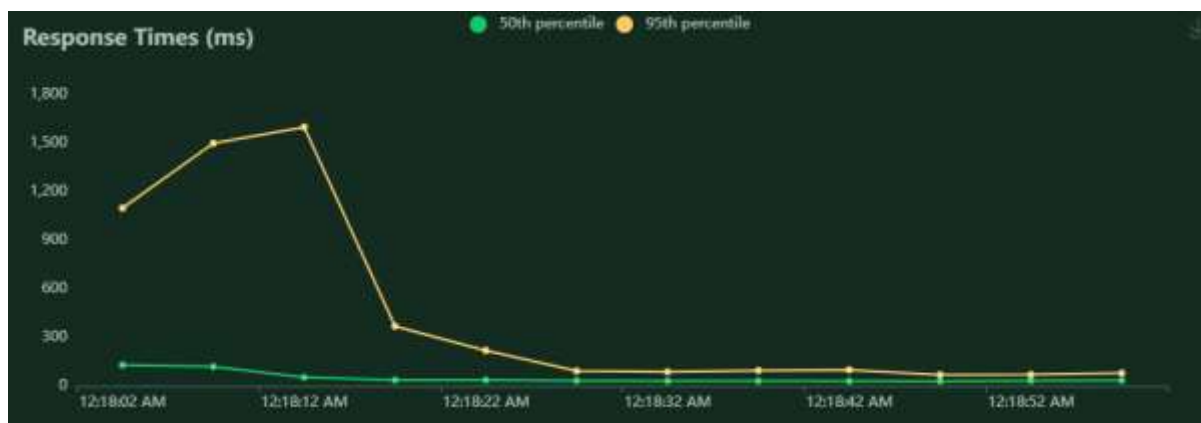
Statistika zahteva tokom dobavljanja uređaja u okviru kuće

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	/api/SmartDevice/GetSmartDevicesForHome/f901e6d3-9fba-4506-a23a-06c91ab97a277?PageNumber=1&PageSize=10	34	39	49	61	87	130	320	850
POST	/api/User/login	1300	1500	1500	1600	1700	1900	2900	2900
Aggregated		35	40	51	66	110	280	1500	2900

Statistika odziva sistema tokom dobavljanja uređaja u okviru kuće



Grafik zahteva i padova u sekundi tokom vremena tokom dobavljanja uređaja u okviru kuće



Grafik vremena odziva sistema tokom vremena tokom dobavljanja uređaja u okviru kuće

Dobavljanje podataka o EV punjaču u pametnoj kući

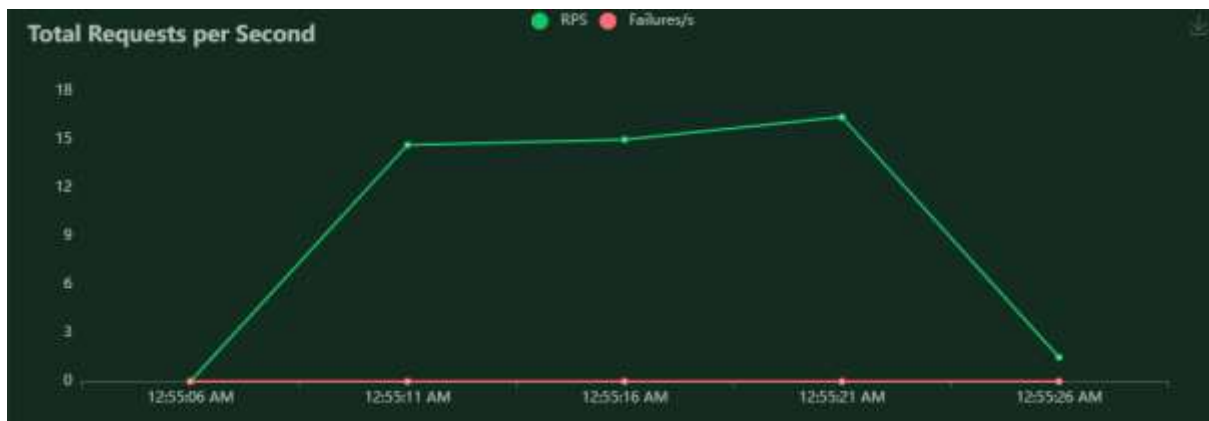
Testiranje dobavljanja podataka o EV punjaču je osnovna operacija u radu sa EV punjaču u pametnoj kući. Test se sastoji od prijavljivanja korisnika, nakon čega se šalje GET zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju EV punjača. Najveći izazov ovog zahteva za server predstavlja dobavljanje podataka iz dva različita perzistentna skladišta. Potrebno je dobiti podatke o svim utičnicama koji imaju manju frekvenciju promene tokom vremena kao i jačinu EV punjača iz PostgreSQL baze, kao i trenutne podatke o kapacitetu svih utičnica, granici punjenja svih utičnica, statusu punjenja svih utičnica i vremenima početka i kraja punjenja svih utičnica kao i podatke o inicijalnom kapacitetu svih utičnica iz time-series Influx baze podataka. Nakon testiranja sistema dobili smo uspešnost od 99% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
POST	/api/User/login	80	0	2087	1179	5663	23	3.8	0.0
GET	/api/VehicleCharger/Get?id=2b8eac46-868b-445e-bb7f-e2e9e2a6e153	88	2	1146	84	13674	1061	4.2	0.1
Aggregated		168	2	1594	84	13674	566	8.0	0.1

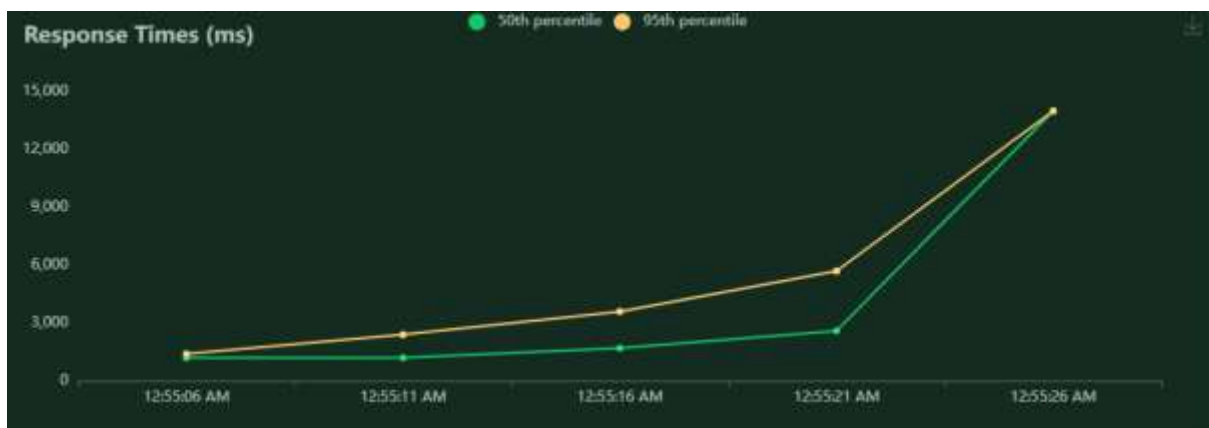
Statistika zahteva tokom dobavljanja podataka o EV punjaču

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
POST	/api/User/login	1900	2100	2300	2600	3000	3600	5700	5700
GET	/api/VehicleCharger/Get?id=2b8eac46-868b-445e-bb7f-e2e9e2a6e153	390	500	1100	1600	2200	4000	14000	14000
Aggregated		1400	1700	2000	2200	2900	3900	14000	14000

Statistika odziva sistema tokom dobavljanja podataka o EV punjaču



Grafik zahteva i padova u sekundi tokom vremena tokom dobavljanja podataka o EV punjaču



Grafik vremena odziva sistema tokom vremena tokom dobavljanja podataka o EV punjaču

Dobavljanje potrošnje po gradu

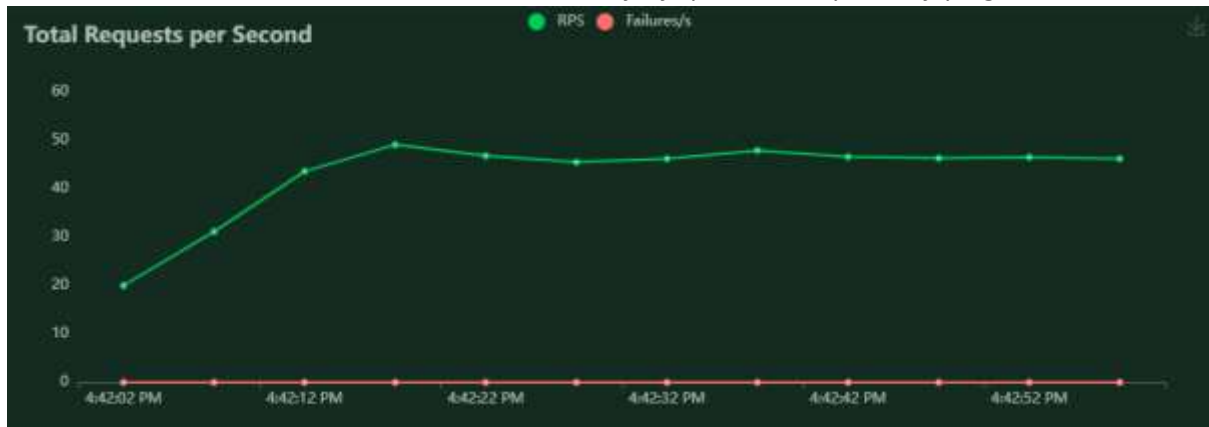
Testiranje dobavljanja podataka o potrošnji svih nekretnina. Test se sastoji od prijavljivanja korisnika, nakon čega se šalje GET zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju grada za koji je potrebno dobiti podatke. Najveći izazov ovog zahteva predstavlja potencijalno dobavljanje velike količine podataka iz time-series Influx baze podataka kao i agregacije velike količine podataka na serveru. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/City/GetCityHistoricalData?id=43ef2fe2-750b-4889-9f3f-12155a0e8e92&from=2023-01-01%2000:00:00&to=2023-12-31%2000:00:00	2619	0	126	28	784	2	43.7	0.0
POST	/api/User/login	100	0	547	296	1250	23	1.7	0.0
Aggregated		2719	0	142	28	1250	2	45.3	0.0

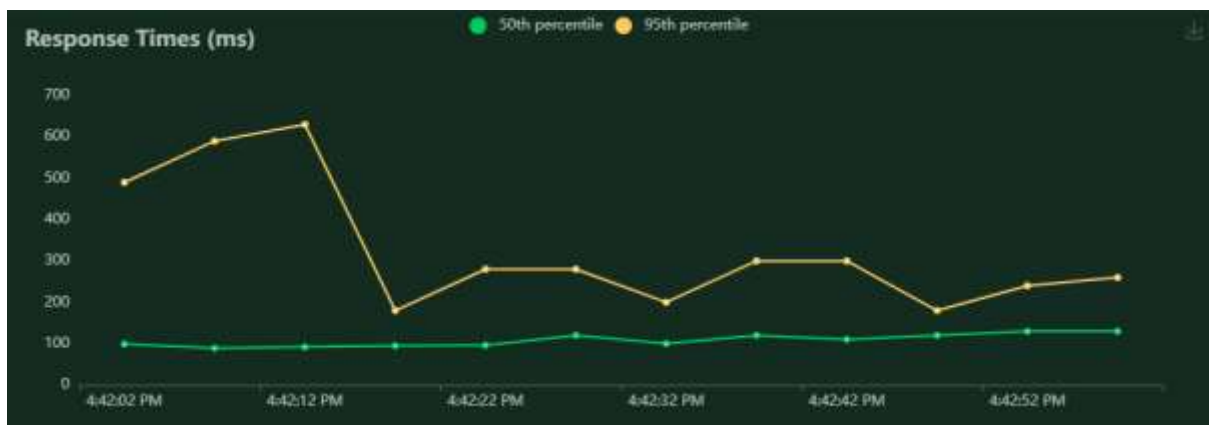
Statistika zahteva tokom dobavljanja podataka o potrošnji po gradu

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	/api/City/GetCityHistoricalData?id=43ef2fe2-750b-4889-9f3f-12155a0e8e92&from=2023-01-01%2000:00:00&to=2023-12-31%2000:00:00	110	120	140	170	200	260	460	780
POST	/api/User/login	480	590	650	680	770	940	1300	1300
	Aggregated	110	130	150	180	230	370	670	1300

Statistika odziva sistema tokom dobavljanja podataka o potrošnji po gradu



Grafik zahteva i padova u sekundi tokom vremena tokom dobavljanja podataka o potrošnji po gradu



Grafik vremena odziva sistema tokom vremena tokom dobavljanja podataka o potrošnji po gradu

Dobavljanje svih gradova

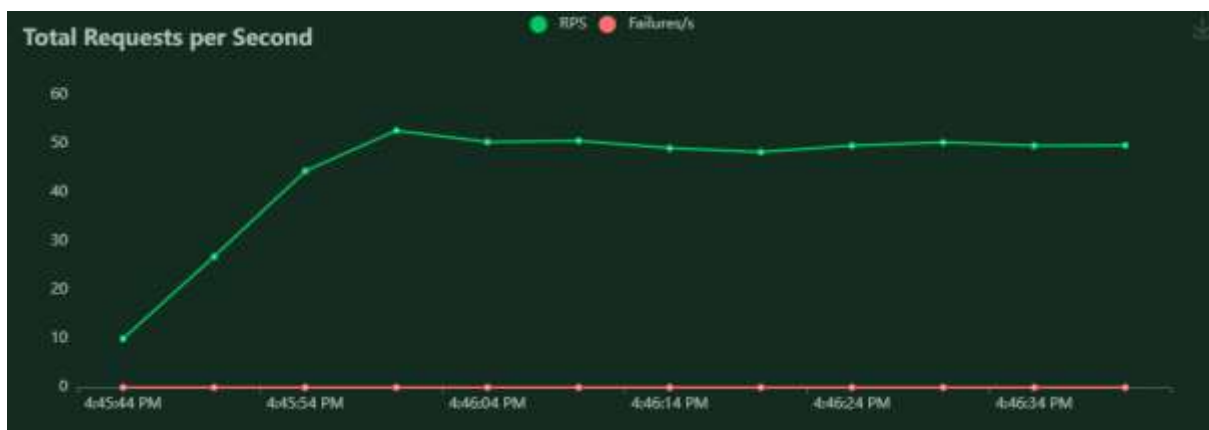
Test se sastoji od prijavljivanja korisnika, nakon čega se šalje GET zahtev ka serveru za dobavljanje svih gradova, bez dodatnih parametara. Napredak koji je značajno ubrzao rad ovog test slučaja je paginacija filtriranih uređaja na backend-u čime se u velikom broju smanjila količina podataka koja se šalje preko mreže i keširanje sadržaja pomoću redis-a. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	/api/City/GetAllCitiesPaged? pageNumber=1&pageSize=10&search=%22%22	2785	0	19	4	117	340	46.4	0.0
POST	/api/User/login	100	0	368	284	486	23	1.7	0.0
Aggregated		2885	0	31	4	486	329	48.1	0.0

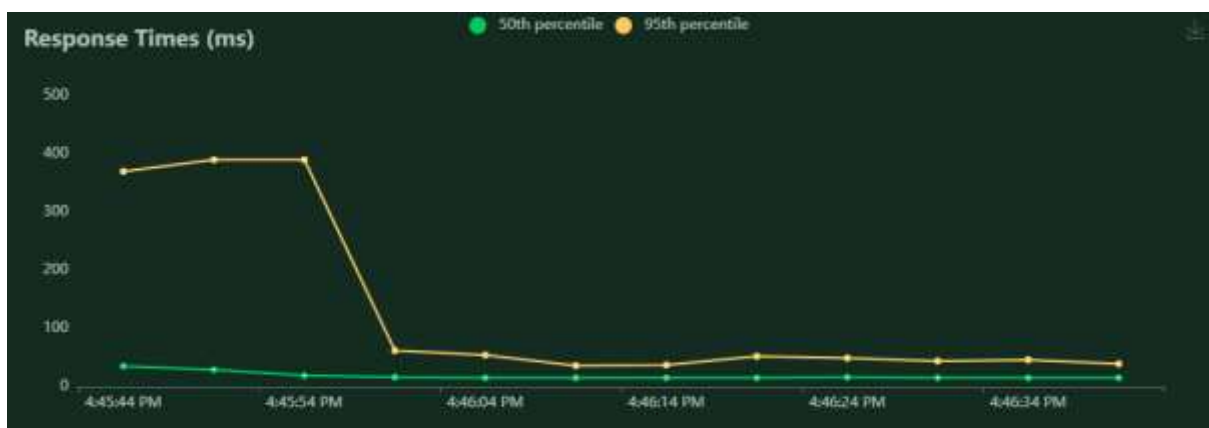
Statistika zahteva tokom dobavljanja podataka o svim gradovima

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	/api/City/GetAllCitiesPaged? pageNumber=1&pageSize=10&search=%22%22	15	17	20	26	37	49	65	120
POST	/api/User/login	390	390	390	400	440	450	490	490
Aggregated		15	17	21	28	44	63	390	490

Statistika odziva sistema tokom dobavljanja podataka o svim gradovima



Grafik zahteva i padova u sekundi tokom vremena tokom dobavljanja podataka o svim gradovima



Grafik vremena odziva sistema tokom vremena tokom dobavljanja podataka o svim gradovima

Dobavljanje istorijskih podataka o akcijama nad solarnim panelom

Testiranje dobavljanja podataka o istoriji akcija solarnog panela je ključna operacija vezana za sigurnost pametnog doma. Test se sastoji od prijavljivanja korisnika, nakon čega se šalje GET zahtev ka serveru pri čemu mu se šalju podaci o identifikacionom broju solarnog panela kao i vremenski opseg za koje je potrebno dobiti podatke. Najveći izazov ovog zahteva za server predstavlja potencijalno dobavljanje velike količine podataka iz time-series Influx baze podataka. Ovaj problem je rešen uvođenjem određenog opsega iz kog se dobavljaju podaci, kao i ograničavanjem maksimalnog opsega na 30 dana. Nakon testiranja sistema dobili smo uspešnost od 100% slučajeva.

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RP5	Failures/s
GET	/api/SolarPanelSystem/GetActionHistoricalData?id=6452026e-aa69-436b-8e53-c77b29c496&from=2023-01-01%2000:00:00&to=2023-12-31%2000:00:00	2754	0	24	12	165	2	45.8	0.0
POST	/api/User/login	100	0	609	504	803	23	1.7	0.0
Aggregated		2854	0	44	12	803	2	47.5	0.0

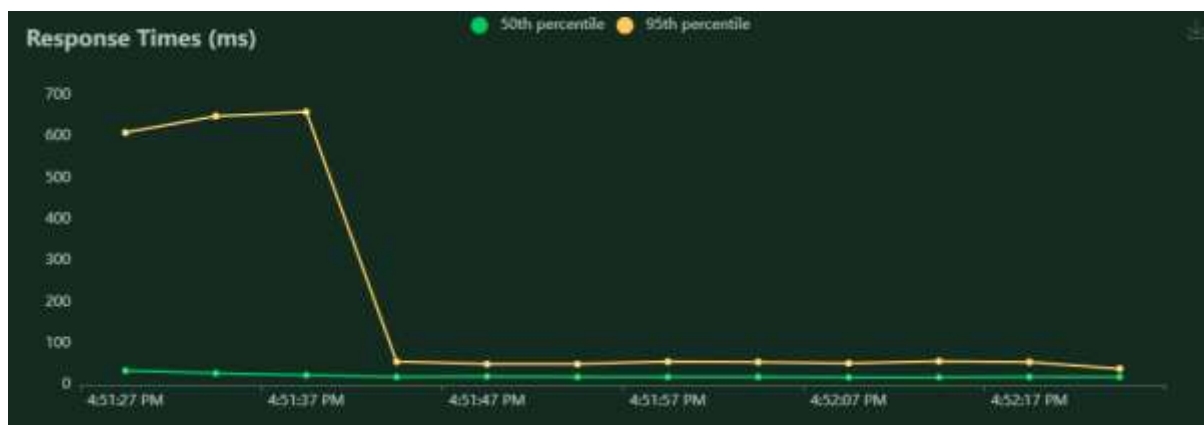
Statistika zahteva tokom dobavljanja istorijskih podataka o akcijama nad solarnim panelom

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	98%ile (ms)	100%ile (ms)
GET	/api/SolarPanelSystem/GetActionHistoricalData?id=6452026e-aa69-436b-8e53-c77b29c496&from=2023-01-01%2000:00:00&to=2023-12-31%2000:00:00	19	21	24	28	41	54	77	170
POST	/api/User/login	600	610	640	680	740	760	800	800
Aggregated		20	22	24	30	49	70	640	800

Statistika odziva sistema tokom dobavljanja istorijskih podataka o akcijama nad solarnim panelom



Grafik zahteva i padova u sekundi dobavljanja istorijskih podataka o akcijama nad solarnim panelom



Grafik vremena odziva sistema tokom dobavljanja istorijskih podataka o akcijama nad solarnim panelom

Testiranje performansi sistema

	10 uređaja	100 uređaja	200 uređaja	300 uređaja	500 uređaja
Paljenje/Gašenje uređaja	~10s	~17s	~22s	~36s	~50s
Dobavljanje podataka o uređaju	~10s	~14s	~19s	~32s	~46s
Promena temperature uređaja	~10s	~17s	~23s	~35s	~51s
Promena režima rada uređaja	~10s	~17s	~23s	~35s	~52s
Zakazivanje rada uređaja	~10s	~19s	~25s	~37s	~55s
Dodavanje utičnica/tablica uređaja	~10s	~19s	~24s	~38	~54s
Otvaranje/Zatvaranje kapije	~10s	~16s	~23s	~36s	~52s

U okviru procesa testiranja performansi sistema simulacije uređaja za pametnu kuću, analizirane su performanse kroz izvođenje sedam različitih akcija sistema, uz varijacije u broju uređaja prisutnih u bazi podataka. Testovi su obuhvatili korišćenje sistema sa 10, 100, 200, 300 i 500 uređaja, koje smo koristili kao markere za merenje vremena odziva. Tokom analize, primećen je eksponencijalni porast vremena odziva sistema, naročito u poslednje dve epohe testiranja, kada je broj uređaja u bazi podataka dostigao 300 i 500. Ova tendencija ukazuje na značajno opterećenje MQTT brokera koji je identifikovan kao glavni uzrok povećanja vremena odziva, a pored njega postoje i ograničenja u vidu mašina na kojima je sistem testiran. Dobijeni rezultati su nam pružili dublji uvid u skalabilnost sistema, na osnovu kojih možemo da utvrdimo potencijalna uska grla unutar pojedinih komponenti sistema.