|  |
| --- |
| FTN Napredne veb tehnologije |
| Testiranje Opterecenja |
| Student 3 |
|  |
| **Nebojsa Vuga** |
| **SV53/2020** |

|  |
| --- |
|  |

Sadrzaj

[Testiranje opterećenja sistema (load testing) za česte scenarije korišćenja 2](#_Toc157886853)

[Testirani slucajevi 2](#_Toc157886854)

[Testiranje kreiranja pametnog uredjaja 2](#_Toc157886855)

[Testiranje upravljanja sistemom solarnih panela 6](#_Toc157886856)

[Testiranje upravljanja elektricnim punjacem za vozila 7](#_Toc157886857)

[Testiranje potrosnje elektricne energije pametne nekretnine 9](#_Toc157886858)

[Testiranje potrosnje/proizvodnje elektricne energije u nekoj nekretnini ili na teritoriji celog grada 11](#_Toc157886859)

[Zakljucak za testiranje opterecenja 12](#_Toc157886860)

[Testiranje perfomansi sistema usled povecanja broja uredjaja koji komuniciraju sa platformom 13](#_Toc157886861)

[Testiranje 14](#_Toc157886862)

[Sistem Solarnih Panela 14](#_Toc157886863)

[Baterija 17](#_Toc157886864)

[Punjac za Elektricna Vozila 19](#_Toc157886865)

[Zakljucak za testiranje perfomansi sistema prilikom rada sa simulatorima 21](#_Toc157886866)

[Napomena: Popravka Resenja 21](#_Toc157886867)

# Testiranje opterećenja sistema (load testing) za česte scenarije korišćenja

Za testiranje opterecenja sistema je koriscena locust biblioteka u pythonu. Testirane su funkcionalnosti implementirane za studenta 3 koje ukljucuju kreiranje pametnih uredjaja, kontrola solarnih panela I punjaca za vozila, prikaz izvestaja za odredjene uredjaje, kao I adminske funkcionalnosti prikaza proizvodnje/potrosnje elektricne energije u nekoj nekretnini ili na teritoriji celog grada. Testiranje se vrsilo u par etapa, prvo se testiralo sa malim brojem korisnika I zahteva, pa sa malo vecim brojem korisnika I zahteva I zatim se islo do same granice pucanja sistema. Testirano je ukupno 5 funkcionalnosti sistema, medju kojima se nalazi vise slucajeva koriscenja.

Testiranje je vrseno sa pokrenutih 400 simulacionih uredjaja kako bi imali neko “strano” opterecenje ka serveru.

## Testirani slucajevi

1. Testiranje kreiranja pametnog uredjaja
2. Testiranje upravljanja sistemom solarnih panela
3. Testiranje upravljanja elektricnim punjacem za vozila
4. Testiranje potrosnje elektricne energije pametne nekretnine
5. Testiranje potrosnje/proizvodnje elektricne energije u nekoj nekretnini ili na teritoriji celog grada

### Testiranje kreiranja pametnog uredjaja

U ovom testu je pokriveno 3 slucaja. Cilj mi je bio da testiram uredjaje po nekoj njihovoj slozenosti I ulozi u sistemu. Testirani uredjaji su bili lampa, sistem solarnih panela i klima uredjaj. Lampa predstavlja najjednostavniji uredjaj od 3, klima predstavlja uredjaj srednje slozenosti, jer ima niz rezima rada, dok sistem solarnih panela predstavlja najslozeniji uredjaj zbog niza panela koji su objekti sa svojim poljima: velicina I efikasnost.

Na pocetku svakog testa je izvrseno logovanje u sistem radi dobijanja jwt tokena koji nam treba za autorizaciju, kao I dobijanje pametne nekretnine kako bi imao njen ID koji koristimo pri kreiranju pametnog uredjaja.

Na sledecim slikama se mogu videti rezultati testova u 4 etape.



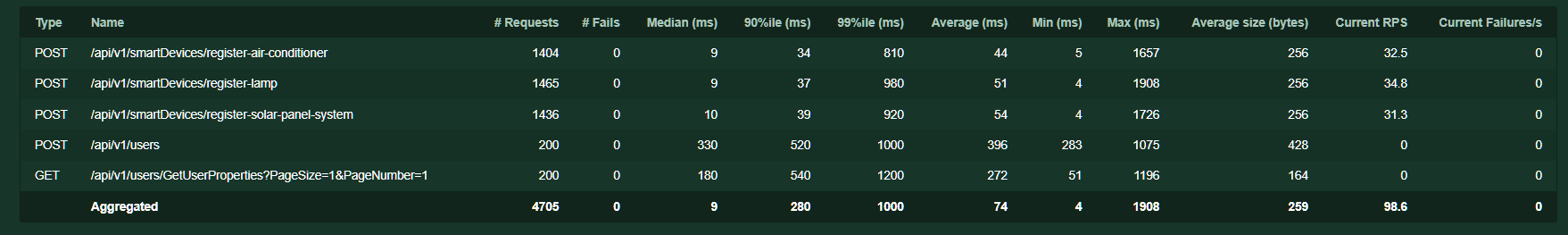
Pokretanje testa sa 10 usera I 2 spawn rate.

Kao sto se vidi sa slike, vreme izvrsavanja zahteva je izuzetno malo, neka promasaja, svi zahtevi su prosli u dobrom vremenu.



Pokretanje testa sa 50 usera I 10 spawn rate.

Idalje je vreme izvrsavanja zahteva za kreiranje uredjaja izuzetno nisko, ali vidi se povecanje vremena u samom logovanju u sistem. Idalje nema promasaja zahteva.



Pokretanje testa sa 200 usera I 20 spawn rate.

Kao I na prethodnoj slici. Kreiranje uredjaja nema problem skaliranja, dok logovanje prihvata velik udarac, I duplirano je vreme trajanja logovanja u sistem u odnosu na logovanje kad imamo 50 usera.



Pokretanje testa sa 1000 usera I 50 spawn rate.

Tek se na 1000 korisnika koji pokusavaju da registruju uredjaj primeti razlika u vremenu, I to velika. Vidimo sad da je prosecno vreme oko 1 sekunde da se uredjaj registruje. Logovanje idalje raste ovaj put na 2.6 sekundi u proseku. Idalje nema promasaja, ali priblizava se tome.



Pokretanje testa sa 2500 usera I 200 spawn rate.

Kao sto vidimo. Na broju korisnika od oko 2500 je doslo do pada sistema. Logovanje u sistem je postalo nenormalno dugacko i sistem to nije mogao ispratiti.

Na sledecoj slici se nalazi grafik na kom mozemo videti kako je sistem reagovao na razlicite nivoe opterecenja.



Grafik testa opterecenja

Sa grafika vidimo da je vreme odziva bilo minimalno na malim brojem korisnika I zahteva po sekundi. Tek kada smo presli 500 korisnika pocinje da raste vreme odziva sistema I vec na hiljadu korisnika I 250 zahteva po sekundi pocinje da koci sistem.

Do pucanja sistema je doslo na nekih 2500 korisnika I samim tim vreme odziva je mnogo poraslo.

Sa testova se moze zakljuciti takodje da je kreiranje sistema solarnih panela trebalo malo duze od druga dva uredjaja iz razloga sto kada kreiramo sistem, moramo kreirati I same solarne panele pa imamo dodatne upise u bazu podataka.

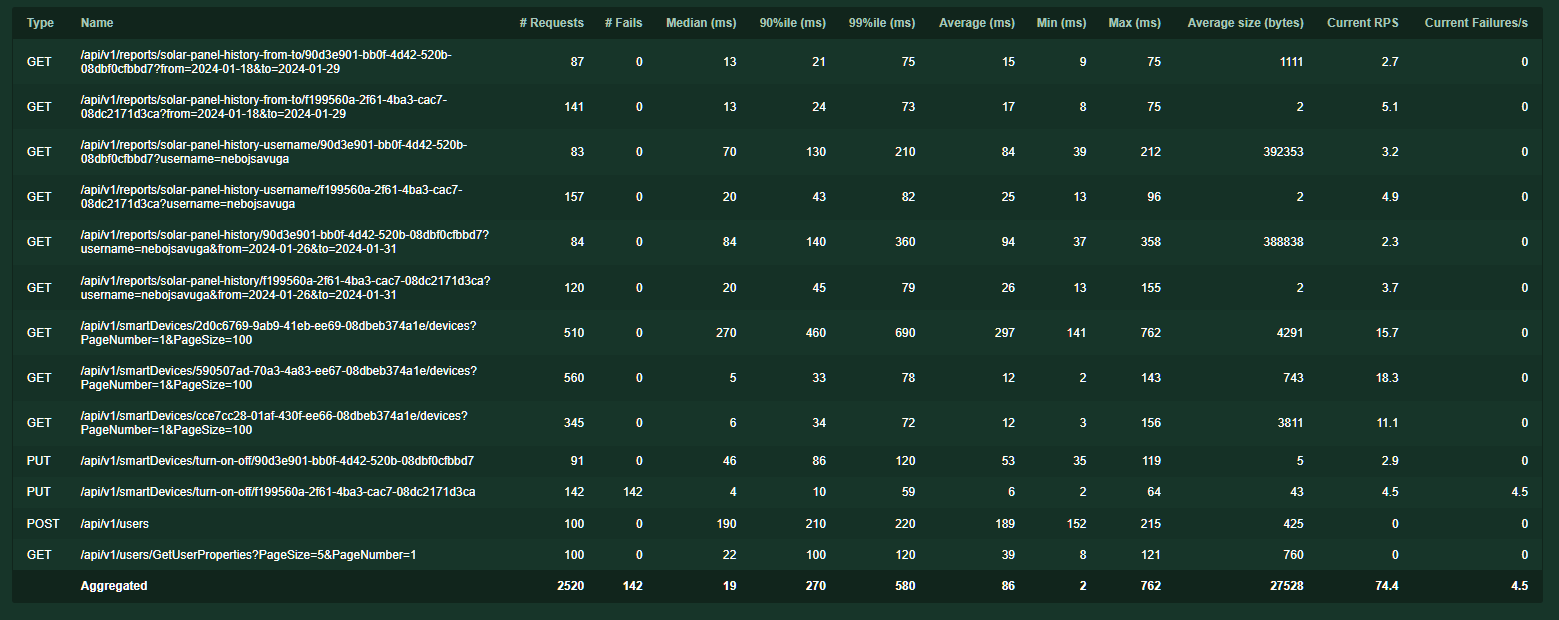
Vidimo da kreiranje samih uredjaja nije bio toliko problem sistemu, nego mu je najveci problem predstavljalo logovanje u sistem koje se vec u prvih par faza videlo da nije najbrze.

### Testiranje upravljanja sistemom solarnih panela

U ovom testu su testirana 4 slucajeva koriscenja, a to su:

* Dobavljanje izvestaja za datum od-do I sa korisnickim imenom
* Dobavljanje izvestaja za datum od-do bez korisnickog imena
* Dobavljanje izvestaja bez datuma sa korisnickim imenom
* Ukljucivanje/iskljucivanje sistema

Na pocetku svakog testa prvo se ulogujemo radi dobijanja tokena I zatim dobavimo property id kako bi mogli pristupiti odredjenoj paneli.



Pokretanje testa sa 100 usera I 10 spawn rate

Preskocio sam prikaz slika sa 10 I 50 usera jer su slike prevelike, a rezultati su isti. Jedan od requestova je failovao jer odredjeni uredjaj nije bio online pa ga nije mogao naci. Vracao je 404 error. Vreme izvrsavanja samih izvestaja je idalje dosta brzo I to samo u par milisekundi.



Grafik testova za sistem solarnih panela

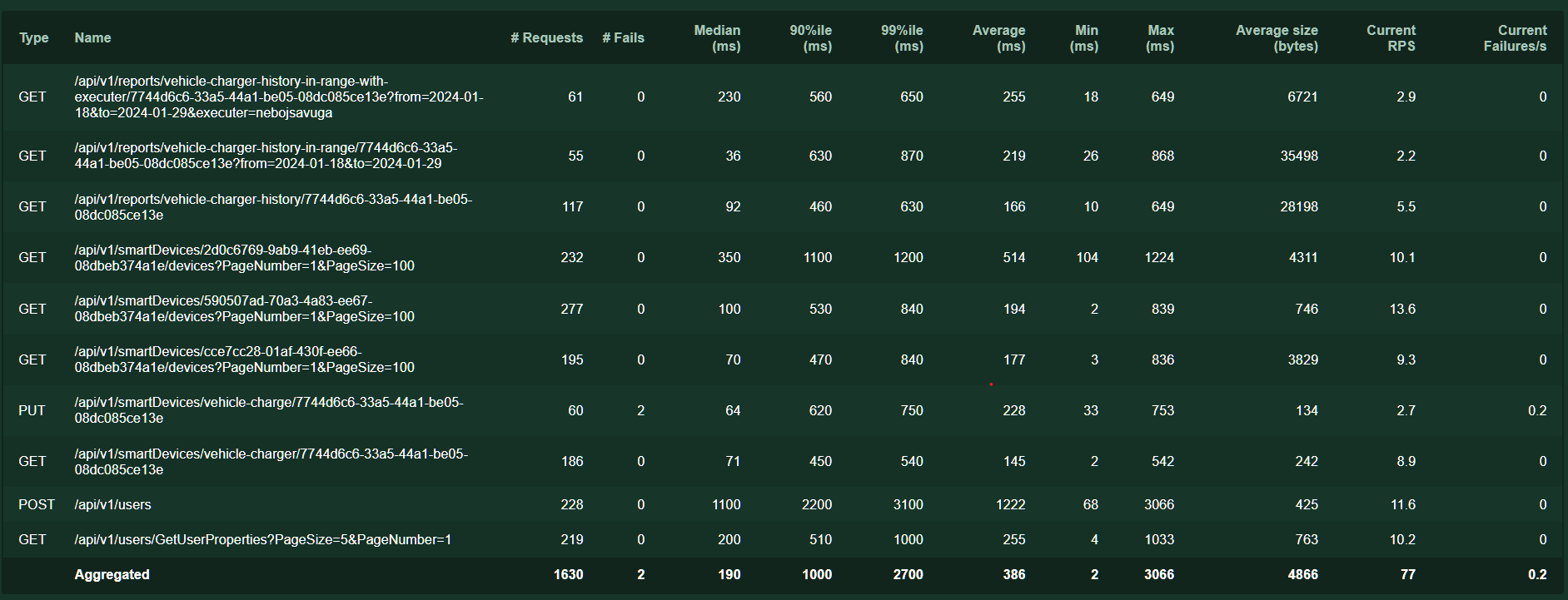
Kao sto se vidi sa grafika, vreme odziva se nije drasticno menjalo do nekih 50 zahteva po sekundi, ili 100 aktivnih usera. Tek sa brojem usera od preko 100 pocinjemo videti drasticnu razliku u vremenu odziva sistema I to u par sekundi u proseku za odgovor. Razlog tome je veca kolicina podataka koja se dobavljala kod testa za ukljucivanje/iskljucivanje sistema. Tu se istovremeno izvrsavao test koji menja to stanje I samim tim se sistem dosta punio tim podacima, koji se u realnosti nece puno menjati, tj sistem solarnih panela u realnom svetu nece biti paljen gasen 100+ puta po sekundi, nego jednom u par dana ako I toliko. Na osnovu tog zakljucka smatram da nije potrebno dodatno optimizovati ovu funkcionalnost, jer realnu svrhu upotrebe izvrsava veoma brzo.

### Testiranje upravljanja elektricnim punjacem za vozila

U ovom testu su testirana 3 slucajeva koriscenja, a to su:

* Dobavljanje izvestaja za datum od-do I sa korisnickim imenom
* Dobavljanje izvestaja za datum od-do bez korisnickog imena
* Promena maksimalnog procenta popunjenosti baterije

Na pocetku svakog testa prvo se ulogujemo radi dobijanja tokena I zatim dobavimo property id kako bi mogli pristupiti odredjenom sistemu za punjenje.



Grafik testova za sistem punjaca elektricnih vozila



Izvestaj testova za sistem punjaca elektricnih vozila

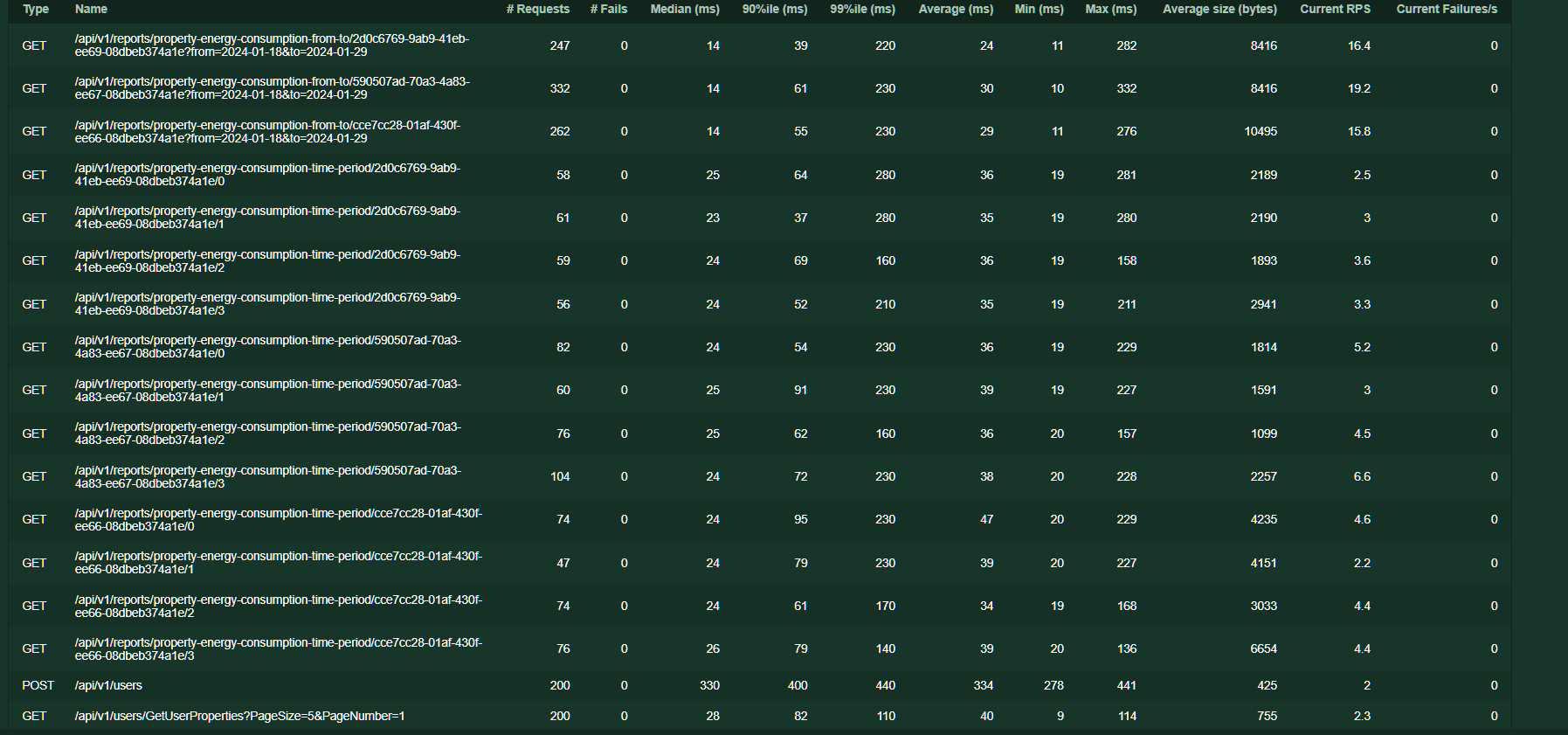
Kako vidimo sa izvestaja vreme izvrsavanja ove funkcionalnosti je nesto sporije nego kod prethodne. Program je poceo da puca kada je broj zahteva dostigao vrednost od oko 70 po sekundi, I to na funkcionalnosti za postavljanje maksimalnog procenta punjenja. Kao I u prethodnom uredjaju, ova funkcionalnost je neka koja se poziva jako retko za odredjeni uredjaj, a posto je ovde izvrsen test sa jednim konkretnim, rezultati su sasvim zadovoljavajuci.

### Testiranje potrosnje elektricne energije pametne nekretnine

Za testiranje ove funkcionalnosti pokrivena su 2 slucaja koriscenja:

* Kada korisnik izabere vremenski period od-do pri cemu taj period ne sme biti veci od mesec dana
* Kada korisnik izabere u poslednjih N casova/dana da se prikaze gde ima opcije 6h, 12h, 24h, 7 dana, 30 dana

Na pocetku svakog testa je dobavljen jwt token I id propertija kao I u svim prethodnim.



Izvestaj testova za potrosnju elektricne energije u

odredjenoj nekretnini, 200 usera, spawn rate 20



Izvestaj testova za potrosnju elektricne energije u

odredjenoj nekretnini, 1000 usera, spawn rate 250

Sistem je dostigao svoj limit I poceo da puca sa nekih 1000 aktivnih korisnika. Razlog tome je sto je svaki zahtev bio naporan za influx bazu koja je kad je doslo do ogranicenja memorije pukla.



Grafik za testove potrosnje u nekoj nekretnini

Sa grafika vidimo da je vreme odziva bilo nisko do nekih 200 korisnika I 100-150 zahteva po sekundi. Sistem je tek poceo da se opterecuje sa brojem korisnika preko 200 I brojem zahteva preko 150. Sa 1000 korisnika sistem je bio neupotrebljiv jer je vreme odziva dostiglo nekoliko sekundi.

### Testiranje potrosnje/proizvodnje elektricne energije u nekoj nekretnini ili na teritoriji celog grada

Kod ovog testa su izvrsena 2 slucaja:

* Kada dobavljamo proizvodnju/potrosnju za jednu nekretninu
* Kada dobavljamo proizvodnju/potrosnju za neki grad

Radi sto veceg opterecenja testovi su paralelno izvrsavani.



Grafik za testove potrosnje u nekoj nekretnini/gradu

Sa grafika vidimo da je vreme odziva bilo nisko I pri 100 zahteva po sekundi. Neka promena se tek pocela videti sa preko 200 zahteva, ali je ubrzo I pala. Do najveceg rasta je doslo kada je broj korisnika dostigao oko 500, a broj zahteva po sekundi oko 300-400 pri cemu se vreme odziva brzo stabilizovalo I vratilo u neku normalnu.

## Zakljucak za testiranje opterecenja

Kako vidimo sa grafikona I tabela, najveci problem sistemu prave PUT zahtevi kontrole uredjaja, odnosno kada previse korisnika pokusa da kontrolise jedan te isti uredjaj. Sistem je na ovo otporan do nekih 50-100 korisnika I isto toliko zahteva po sekundi, a posto su to funkcionalnosti koje se jako retko koriste u stvarnosti, I retko ce biti izvrsene na istom uredjaju vise puta u sekundi, smatram da je test opterecenja prosao koliko toliko uspesno.

# Testiranje perfomansi sistema usled povecanja broja uredjaja koji komuniciraju sa platformom

Ovaj test je uradjen tako sto je registrovano par hiljada uredjaja odredjenog tipa I onda je za svaki tip pokrenut odredjeni broj simulatora do pucanja sistema. Server se sam po sebi nije gasio/pucao I nije previse opterecivao, ali je mqtt brooker imao problem sa brojem uredjaja od oko 2000.

Memorijski nismo mogli ni testirati sa vise uredjaja s tim da samo doker I bek kad se pokrenu, zauzmu oko 85% memorije, pa nismo imali mnogo lufta za jos veci test.

Problem je bio taj da nestaje konekcija sa brookerom. Na pocetku je radilo sa 300 uredjaja maksimalno, pa zatim sa 500, pa do nekih 2000. Do ovog napretka smo dosli tako sto smo simulatore pokretali u razlicito vreme. Npr, nece svaka baterija poslati svoj status na 10 sekundi, nego ce neka poslati na 8, neka na 11 itd… Samim tim smo smanjili broj istovremenih zahteva sa 2000 na 300 400.

U daljem tekstu ce biti prikazani odzivi sistema prilikom povecanja broja uredjaja koji salju/primaju poruke od/ka server. Bice prikazani testovi koji ukljucuju ukljucivanje/iskljucivanje uredjaja prilikom rada odredjenje kolicine istih, postavljanje vrednosti procenta punjenja punjaca, kao i zahtevi za izvestaje za odredjene uredjaje da bi se videlo kako taj velik broj utice na bazu sistema.

## Testiranje

Pre testa svakog uredjaja prvo je generisano oko 2000 uredjaja tog tipa I pritom su pokrenuti par uredjaja drugih tipova kako sistem samo ne bi radio sa jednim tipom uredjaja nego sa vise. Znaci ukoliko testiram panelu, palim sa 500, 1000… panela I uvek imam u pozadini da se vrti I par svih drugih tipova uredjaja radi realnosti testa.

Svaki test ce se izvrsiti tako sto ce se dobavljati get zahtevi za neke izvestaje za uredjaj testiranog tipa, kao I put requestovi za ukljucivanje/iskljucivanje uredjaja I postavljanje maksimalne vrednosti punjenja za punjac.



Kreirano ~2000 panela za testiranje panela

### Sistem Solarnih Panela

Testovi su pokrenuti u 3 etape:

* Sa 100 SPS uredjaja
* Sa 500 SPS uredjaja
* Sa 1000 SPS uredjaja

Pri svakoj etapi je testirano sa povecanjem broja korisnika I zahteva po sekundi.

Testirano je tako da se salju GET zahtevi uporedo sa PUT zahtevima za odredjeni uredjaj.

Rezultati su sledeci:



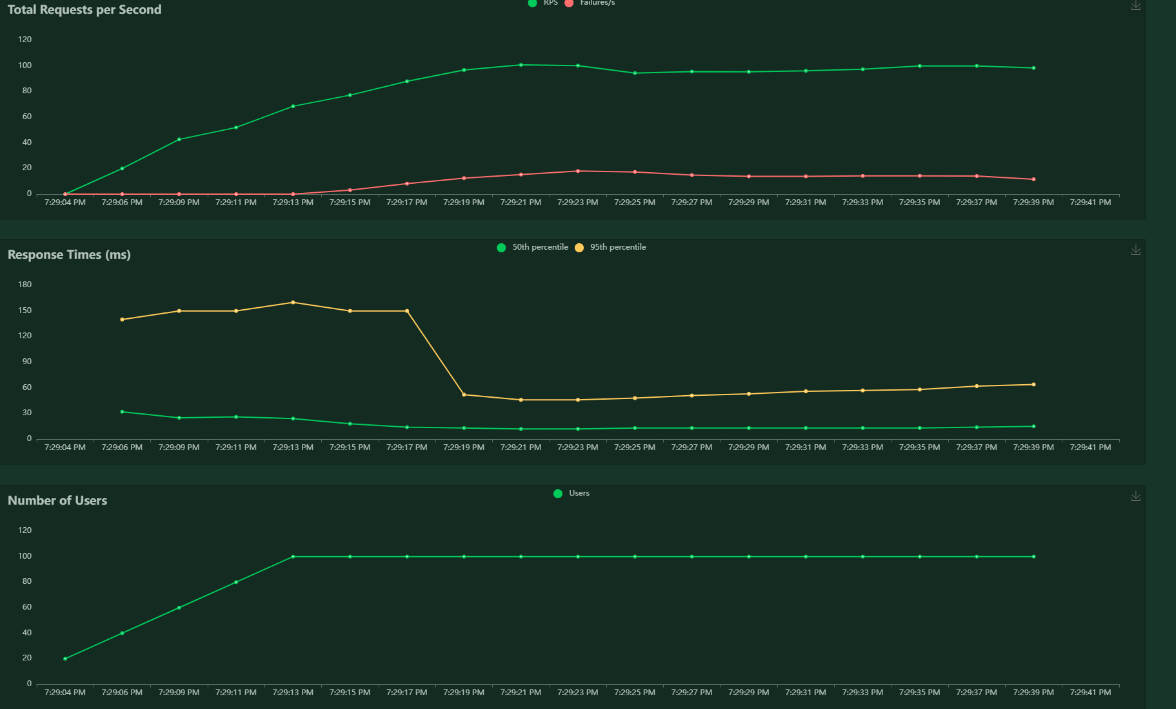
100 Simulatora I do 250 zahteva po sekundi

U prvoj fazi je pokrenuti 100 simulatora solarnih panela I sistem je dosao do 250 zahteva u sekundi pre nego sto je poceo pucati. Server kao server nije pukao, nego se desavalo kao I u vecini testova da mqtt ne moze da izbarata sa toliko put zahteva I diskonektuje se. Vreme odziva sa ovom kolicinom simulatora je bilo pristojno sa nekih 100/200 ms. Treba uzeti u ozbir da najvece vreme odziva oduzima login funkcionalnost.



500 simulatora I do 200 zahteva po sekundi

U drugoj fazi testa je pokrenuti 500 simulatora I dostiglo je 200 zahteva u sekundi pre nego sto je mqtt brooker poceo da diskonektuje. Vreme odziva je u nekim momentima cak bilo I nize nego sa 100 simulatora, ali kao sto mozemo videti broj mogucih zahteva u sekundi se smanji za 20%.



1000 simulatora I do 120 zahteva po sekundi

U trecoj fazi je pokrenuto 1000 simulatora I nije moglo preci preko 100 zahteva u sekundi pre nego sto je sistem poceo da luduje. Ovaj put greske nisu bile ni mqtt tipa nego je baza bila preopterecena I nije mogla da pronadje uredjaje u nekim momentima. Vreme odziva se opet nije menjalo sto znaci da broj simulatora solarnih panela ne utice na brzinu odziva sistema.

### Baterija

Kao I u prethodnom testu, prvo je kreiramo ~2000 baterija na 3 korisnika.

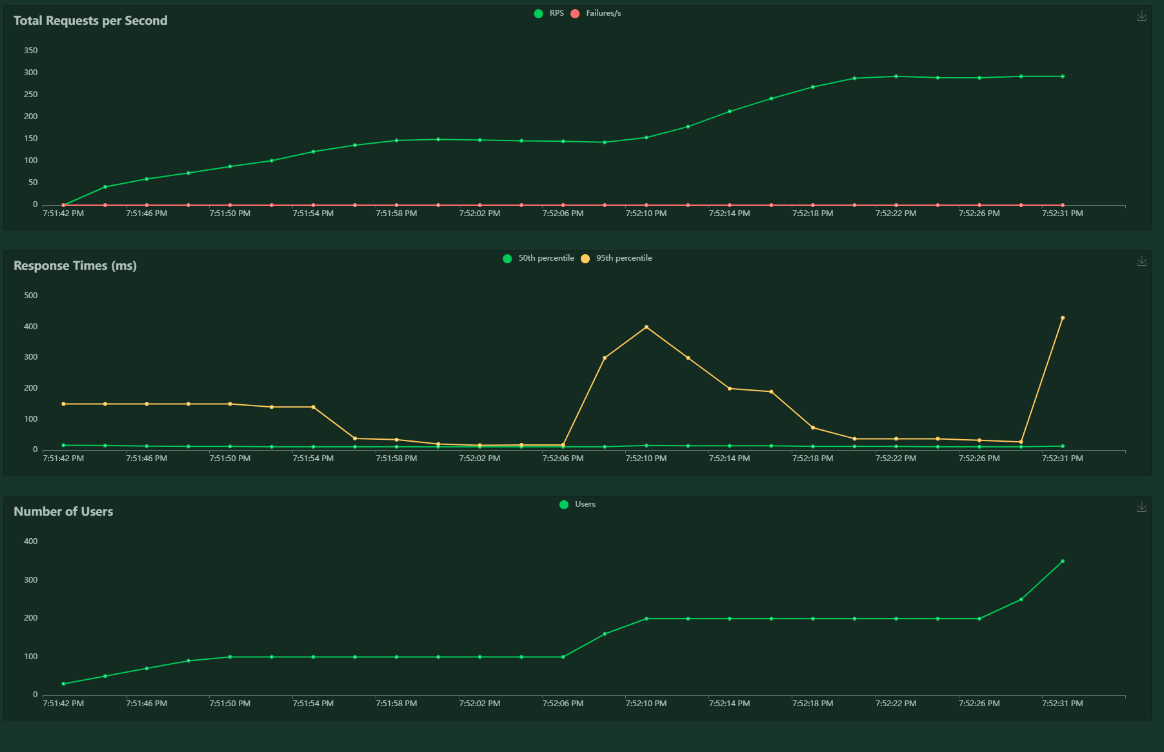


Kreiranje 2000 baterija



100 baterija I 700 zahteva po sekundi

U prvoj fazi testa je korisceno 100 simulatora I sa 700 zahteva po sekundi se moze reci da se sistem sasvim solidno drzao. Skok odziva sistema se pojavio tek posle 200-300 zahteva po sekundi I nikad nije prekoracio 1 sekundu za odziv.

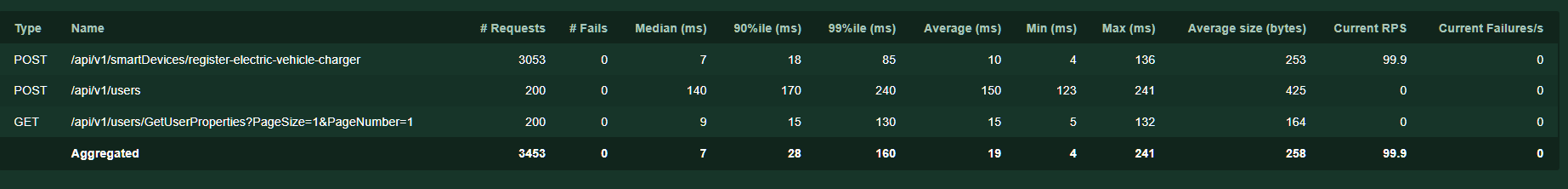


1000 simulatora I 300 zahteva u sekundi

Sa 1000 simulatora se takodje sistem drzao sasvim solidno sa ponekim skokom vremena odziva ali opet ne preko jedne sekunde. Broj zahteva je dostigao 300 I doslo je do preopterecenja influx baze podataka, CPU iskoriscenje za influx je prelazilo preko 100% I naravno sistem vise nije mogao da dobavi podatke.

### Punjac za Elektricna Vozila

Kao i u prethodnim testovima prvi korak je kreiranje odredjenog broja uredjaja.



Kreiranje oko 3000 punjaca

Zatim isto kao I u prethodnim, testira se sistem sa 100 aktivnih simulatora, 500 I 1000 aktivnih simulatora.

Nazalost mqtt nije mogao da izdrzi vise od 20 zahteva po sekundi, pa su rezultati sa pokrenutih 100, 500 I 1000 uredjaja identicni.



100 simulatora I 40 zahteva po sekundi

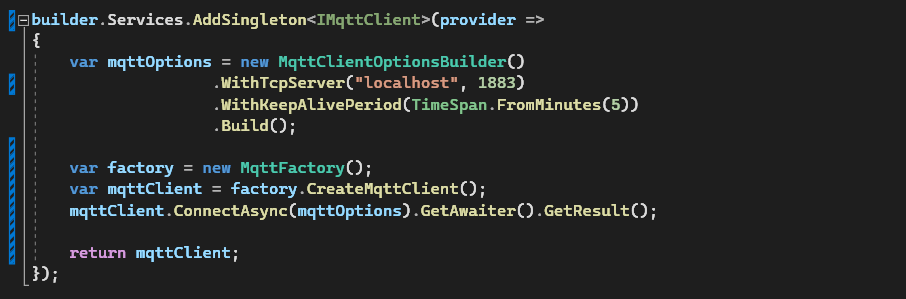
Mqtt je vec sa 20ak zahteva po sekundi poceo da se diskonektuje pri promeni maksimalne popunjenosti baterije kod punjaca. Ovo je funkcionalnost koja se svakako ne poziva cesto, i za scenario rada kojim bi radila se pokazala dovoljno dobra.

## Zakljucak za testiranje perfomansi sistema prilikom rada sa simulatorima

Sa testiranja mozemo videti da su problem sistemu zadavale komponente na koje mi ne mozemo previse uticati kao sto su mqtt protocol I influx baza. Resenje za influx bazu bi bio povecanje procesorske moci I memorije, a za mqtt koriscenje mozda neke bolje opcije kao sto su maximum message rate. Pokusao sam da odrzim konekciju sa mqttom sto je duze moguce u postavci programa, ali to nije pomoglo pri resavanju problema.

# Napomena: Popravka Resenja

Sa velikim trudom I naporom smo kolega Bogdan Janosevic I ja skontali gde nam je bio problem sa mqtt brookerom.Instacirali smo ga u vise servisa I cesto je gubio/dobijao konekciju, tj u svakom servisu se konektovao. Ovaj problem smo resili tako sto smo ga konektovali samo kad se pokrece aplikacija(Program.cs) kao singltoj I dalje ga prosledjivali kao parameter konstruktora u servisima.

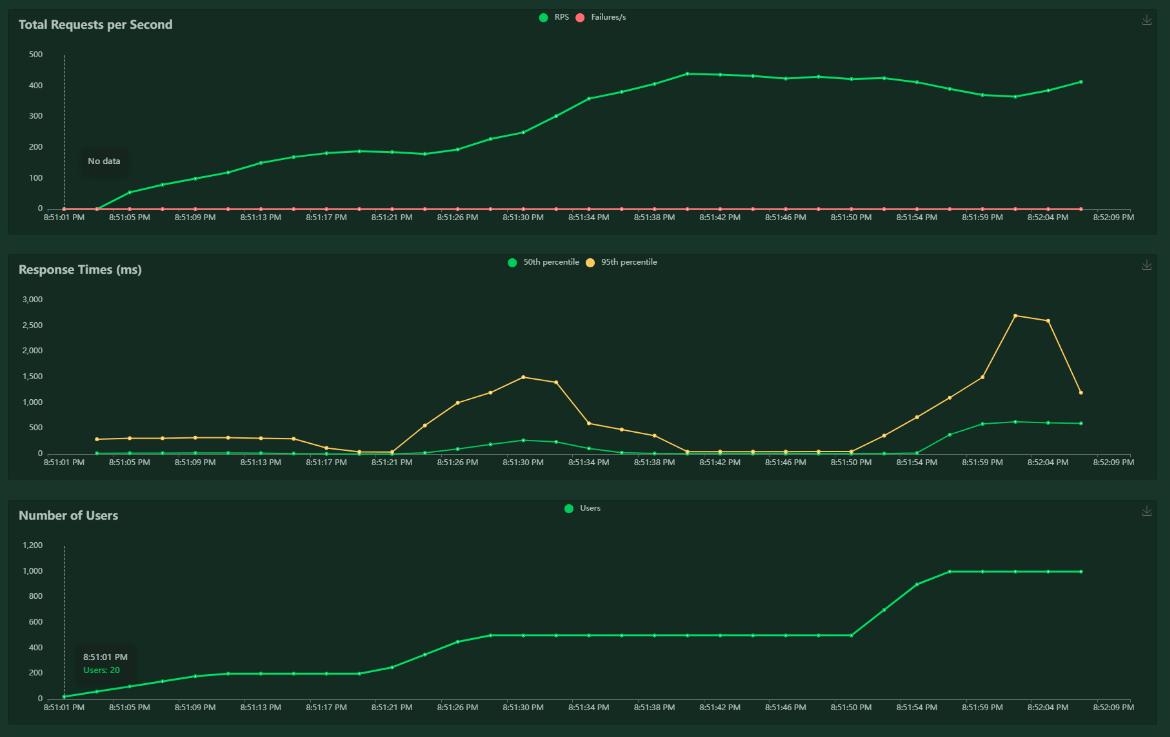


Resenje sa mqtt brookerom

U nastavku dajem prilog dva izvestaja koja su se bavila testiranjem sa PUT zahtevima promenom stanja simulatora, izvestaju si za charger za max charge rate I za sps ukljucivanje/iskljucivanje.



Grafici za solarne panele 1000 korisnika I 300 RPS



Grafik za charger 1000 Korisnika I 500 RPS

Kao sto vidimo sa slike, vise nema pucanja u sistemu, RPS je mnogo veci I prosecno vreme odziva je idalje relativno nisko sa velikim brojem zahteva I korisnika. Nije bilo moguce testirati sa vise korisnika zbog racunarskih resursa.

Pre ove popravke je radilo sa oko 30-40 RPS za charger I 100 RPS za sistem solarnih panela I sa 200 usera, a sad radi bez problema sa 1000 korisnika I 500 RPS.

Zbog spremanja sistema za odbranu nismo u mogucnosti da pokrenemo sve testove ponovo kako se ne bi kreiralo/brisalo po sistemu.