

5 DANA U OBLACIMA

ZADATAK

Nalazite se u ulozi software inženjera kompanije Levi9 koja razvija novu Cloud platformu – Levi9 Cloud.

Zaduženi ste za izradu dela sistema koji računa troškove korišćenja Cloud platforme. Na vama je da implementirate API koji prihvata od Cloud servisa informacije o izvršenim akcijama i algoritam koji na osnovu cenovnika računa ukupnu cenu i cenu po pojedinačnom servisu.

Servisi koje Levi9 Cloud-a podržava su:

- FUNC - Serverless funkcija
- DB - Baza podataka
- OS - Skladište objekata
- VM - Virtuelna mašina
- NETWORK – Mreža

Moguće akcije koje Levi9 Cloud pricing algoritam prihvata nad servisima su:

- Serverless funkcija - FUNC
 - EXEC - Izvršavanje sa opcionim sadržajem
- Baza podataka - DB
 - INSERT - Upis podataka
 - SELECT - Čitanje podataka
 - SOFT_DELETE - Brisanje podataka
- Skladiste objekata – Object Storage (OS)
 - PUT - Dodavanje objekta
 - GET – Preuzimanje objekta
 - SOFT_DELETE - Brisanje objekta
- Virtuelna mašina - VM
 - START - Startovanje virtuelne mašine
 - STOP - Zaustavljanje virtuelne mašine
- Mreža - NETWORK
 - N/A - Indirektni prenos podataka

Nad DB i OS je uključen versioning. Akcija SOFT_DELETE ne uklanja podatke sa diska već ih logički briše. Za podatke koji su jednom kreirani, biće naplaćen storage cost i nakon SOFT_DELETE-a.

VM servis je trenutno u Alfa fazi implementacije i kao takav podržava samo jednu instancu virtuelne mašine u određenom trenutku. To znači da se druga instanca ne može startovati dok se prethodna ne stopira.

Izvršenje nekih od akcija može izazvati indirektno izvršenje drugih akcija. Ovo se odnosi na prenos podataka kroz mrežu.

Levi9 Cloud naplaćuje usluge u Dolarima.

Način naplate usluga Levi9 Cloud-a može biti, u zavisnosti od usluge:

- Na osnovu broja izvršavanja
 - 0.01 USD na svakih zapocetih 10 izvršavanja
 - Besplatni limit: 10 izvršavanja
- Vremenski
 - 0.01 USD za svaku sekundu
 - Besplatni limit: 10 sati
- Po jedinici kolicine podataka
 - 0.01 USD za svaki MB
 - Besplatni limit: 1GB

Način naplate po tipu servisa je sledeći:

- Serverless funkcija
 - Na osnovu broja izvršavanja
 - Po jedinici količine podataka prenetih kroz mrežu prilikom izvršavanja funkcije
- Baza podataka
 - Na osnovu broja izvršavanja akcija
 - Po jedinici količine podataka na osnovu ukupne veličine baze u trenutku upita/naplate
- Skladište objekata
 - Po jedinici kolicine podataka na osnovu ukupne veličine svih objekata u trenutku upita/naplate
 - Po jedinici kolicine podataka prenetih kroz mrežu prilikom preuzimanja podataka
- Virtuelna mašina

- Vremenski za sve vreme dok je u startovanom stanju
- Ako je u trenutku naplate virtuelna masina i dalje u startovanom stanju, za naplatu se uzima period od trenutka startovanja do trenutka naplate

Usluge koje imaju besplatni limit su besplatne dok se besplatni limit ne potroši, nakon čega počinju da se naplaćuju. Iznos besplatnog limita je definisan po servisu i poseban za svaki tip servisa.

Na primer: 10 FUNC EXEC i 10 DB INSERT/SELECT/SOFT_DELETE.

Cenovnik usluga u Dolarima, koji sistem mora koristiti, je prikazan ispod i dostupan je na <https://5danauoblacima-docs.s3.eu-central-1.amazonaws.com/levi9-cloud-price-list.json>

```
{
  "invocation": {
    "freeTier": 10,
    "step": 10,
    "pricePerStep": 0.01
  },
  "timeSec": {
    "freeTier": 36000,
    "step": 1,
    "pricePerStep": 0.01
  },
  "dataMb": {
    "freeTier": 1024,
    "step": 1,
    "pricePerStep": 0.01
  }
}
```

Free tier vrednost mora biti deljiva sa step vrednošću. PricePerStep se odnosi na cenu jednog celog stepa. Naplata se vrši isključivo po započetom stepu za ceo započeti step unapred.

Primer formata poruke kojom se inicira izvršavanje određene akcije pricing algoritma je sledeći (payloadSizeMb je opciono polje):

```
[
  {
    "userId": "1",
    "serviceType": "FUNC",
    "actionType": "EXEC",
    "timestamp": 1609500600,
    "payloadSizeMb": 5
  }
]
```

Pod pretpostavkom da su svi besplatni limiti već potrošeni, tj. da je do sada izvršeno tačno 10 EXEC akcija čiji je ukupni payload bio tačno 1GB, računica naplate ove akcije bi bila sledeća:

0.01 USD za sledećih započelih 10 izvršavanja (step 10)

0.05 USD za 5 MB prenetih kroz mrežu (step 1)

0.06 USD je ukupna cena

Aplikacija za računanje troškova korišćenja Levi9 Cloud platforme treba da implementira REST API po OpenAPI specifikaciji koja se nalazi na:

<https://5danauoblacima-docs.s3.eu-central-1.amazonaws.com/levi9-cloud-openapi-spec.json>

Možete je vizualizovati koristeći, na primer, <https://editor.swagger.io/>

Implementirani REST API mora da omogućava izvršavanje akcija tako da može da primi listu **hronološki sortiranih akcija**.

Pomenuti REST API treba da:

- Sadrži endpoint dostupan na: POST */actions*

- Koji prima JSON payload (payloadSizeMb je opciono polje)

```
{
  "serviceType": "VM",
  "userId": 1,
  "actionType": "START",
  "timestamp": 1609500600,
  "payloadSizeMb": 1
}
```

- Vraća HTTP status 200 nakon obrađenog zahteva
- Omogućava proveru trenutnog stanja troškova po jednom useru
REST endpoint treba da:
 - Se nalazi na: *GET /user/{userId}/costs*
 - Zahteva kao path parametar **userId**
 - Zahteva kao query parametar **untilDate** u Unix timestamp formatu (<https://www.epochconverter.com/>)
 - Na primer: *GET /user/1/costs?untilDate=1609502400*
 - Vraća rezultat u JSON formatu

```
{
  "totalCosts": 100,
  "costsPerService": [
    {
      "serviceType": "VM",
      "cost": 50
    },
    {
      "serviceType": "DB",
      "cost": 50
    }
  ]
}
```

- Pomenuti REST endpoint može da sadrži filter: **serviceTypes** u query parametru.
 - Na primer: GET `/user/1/costs?untilDate=1609502400&serviceTypes=VM&serviceTypes=DB`
 - Tada vraća rezultat:

```
{
  "totalCosts": 100,
  "costsPerService": [
    {
      "serviceType": "VM",
      "cost": 50
    },
    {
      "serviceType": "DB",
      "cost": 50
    }
  ]
}
```

TEHNIČKI ZAHTEVI REŠENJA

Tehnologije:

- Obavezno: Java ili Javascript/Typescript
- Obavezno: korišćenje build alata ili package manager-a (Maven/Gradle/NPM/Yarn/...)
- Obavezno: rešenje zadatka mora da radi sa opensource free software rešenjima

Rešenje treba da sadrži:

- Izvorni kod
- Fajlove korišćene za testiranje
- Dokumentaciju u Readme fajlu:
 - Opis okruženja potrebnog da se uradi build
 - Kako se radi build
 - Primer kako se aplikacija pokreće
 - Listu korišćenih tehnologija sa kratkim opisom

Svaki prijavljeni takmičar dobija korisnički nalog na GitLab-u, gde je potrebno kreirati repozitorijum koji će sadržati rešenje zadatka.

Rešenjem se smatra sadržaj kreiranog Git repozitorijuma 9. decembra u 9 časova.

PRAVILA BODOVANJA

- Tehničko rešenje – **40%**
 - Subjektivni kvalitet rešenja i koda - 30%
 - Unit testovi - 5%
 - Readme - 5% – objašnjenje na koji način se builda i pokreće projekat
- Funkcionalni zahtevi - **60%**
 - Test case 1 – 20% (javni test case)
 - Prikazan je u Gherkin tekstu ispod
 - Lista akcija koje test šalje se nalazi na:
 - <https://5danauoblacima-docs.s3.eu-central-1.amazonaws.com/levi9-cloud-public-testcase-actions.json>
 - Očekivani totalCosts za
GET /user/1/costs?untilDate=1638454976 je 44 USD
 - Ostali test case-ovi - 40% (tajni)

Scenario: Total Costs are calculated when all actions are executed outside of free tier

Given service type **FUNC** and action type **EXEC** for userId **1** with payload **120**

And service type **DB** and action type **SELECT** for userId **1** with payload **500** MB

And service type **DB** and action type **INSERT** for userId **1** with payload **1025** MB

And service type **DB** and action type **SOFT_DELETE** for userId **1** with payload **525** MB

And service type **OBJECT_STORAGE** and action type **PUT** for userId **1** with payload **1025** MB

And service type **OBJECT_STORAGE** and action type **GET** for userId **1** with payload **500** MB

And service type **OBJECT_STORAGE** and action type **SOFT_DELETE** for userId **1** with payload **1025** MB

And service type **VM** and action type **START** with timestamp **12 hours** ago for userId **1** with payload **0**

And service type **VM** and action type **STOP** for userId **1** with payload **0**

When request is sent to cloud pricing application with given payload and invocations for **FUNC AND DB outside of free tier**

Then total costs for userId **1** will be calculated until date **1638454976**