

Sistem za upravljanje prodavnicom

Uvod

Osnovni ciljevi projekta su:

- implementacija veb servisa koji čine dati sistem;
- pokretanje sistema pomoću alata za orkestriranje kontejnera;

Sistem je namenjen za višekorisnički rad i stoga treba pažljivo da bude dizajniran za potrebe ispravnog i efikasnog rada. Deo funkcionalnosti sistema je javno dostupan, dok je deo funkcionalnosti obezbeđen samo za korisnike koji mogu da se prijave na sistem.

Sistem je potrebno implementirati korišćenjem `Python` programskog jezika i `Flask` i `SQLAlchemy` biblioteka. Prilikom obrade zahteva korisnika neophodno je koristiti `SQLAlchemy` u što većoj meri, odnosno realizovati obradu pomoću SQL upita svugde gde je to moguće. Pored koda, potrebno je priložiti `Docker Image` šablone koji predstavljaju delove sistema i koji se mogu iskoristiti za pokretanje odgovarajućih kontejnera. Pored implementacije, potrebno je napisati konfiguracione fajlove pomoću kojih se ceo sistem može pokrenuti na klasteru računara.

Konceptualni opis sistema

Sistem treba da obezbedi registraciju korisnika (kupac ili magacioner). Kupci mogu da vrše pretragu i kupovinu proizvoda. Magacioneri su radnici u prodavnici koji ažuriraju trenutno stanje proizvoda (količinu i cenu).

Svaki korisnik treba da bude registrovan pre korišćenja sistema. Za svakog korisnika se u okviru njegovog korisničkog naloga čuvaju sledeće informacije: email adresa i lozinka koje se koriste prilikom prijave, ime, prezime i uloga korisnika. Korisnik može imati ulogu kupca ili magacionera. U sistemu takođe postoji korisnik sa ulogom administratora koji se dodaje prilikom inicijalizacije sistema.

Korisnici sa ulogom magacionera (u nastavku magacioner) mogu da ažuriraju trenutno stanje proizvoda u prodavnici.

Za proizvode se pamte sledeće informacije: kategorija kojoj proizvod pripada (može ih biti više), ime proizvoda, njegova cena i trenutno dostupna količina. Za svaku kategoriju se pamti njeno ime.

Korisnici sa ulogom kupca (u nastavku kupac) mogu da vrše pretragu proizvoda, da prave narudžbine i da vrše pregled napravljenih narudžbina.

Prilikom svake kupovine proizvoda u sistemu se pravi narudžbina. Za svaku narudžbinu se čuva spisak proizvoda, ukupna cena narudžbine, njen status i trenutak njenog kreiranja. Narudžbina može biti uspešna ili na čekanju (u zavisnosti od trenutnog stanja proizvoda u prodavnici).

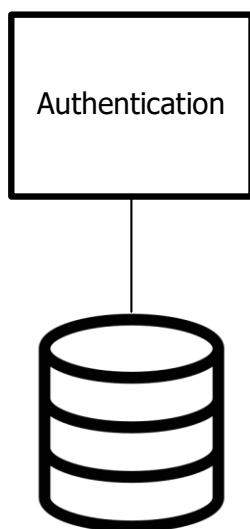
Korisnici sa ulogom administratora (u nastavku administrator) mogu da prave izveštaje o poslovanju prodavnice.

Funkcionalni opis sistema

Sistem se sastoji iz dva dela: jedan namenjen za upravljanje korisničkim nalogima i jedan namenjen za upravljanje prodavnicom. I jedan i drugi deo sistema se sastoje iz nekoliko komponenti koje su realizovane pomoću kontejnera kreiranih na osnovu `Docker Image` šablona. Deo ovih `Docker Image` šablona već postoji i nalazi se u okviru javnog repozitorijuma na adresi <https://hub.docker.com/>. Ostatak je neophodno implementirati. U nastavku je dat funkcionalni opis sistema.

Upravljanje korisničkim nalogima

Izgled dela sistema koji je zadužen za upravljanje korisničkim nalogima dat je na slici 1.



Slika 1. Izgled dela sistema za upravljanje korisničkim nalogima

Ovaj deo sistema se sastoji iz jednog kontejnera i baze podataka u kojoj se nalaze samo podaci o korisnicima. Kontejner predstavlja veb servis pomoću kojeg korisnik može da se registruje i dobije [JSON veb token](#) sa kojim će moći da pristupi ostatku sistema.

Opis funkcionalnosti koje pruža ovaj veb servis dat je u nastavku. Svaka adresa je relativna u odnosu na IP adresu kontejnera i broj porta na kojem kontejner sluša.

- **Registracija korisnika**

Adresa /register

Tip POST

Zaglavlja -

Telo Telo zahteva je JSON objekat sledećeg formata:

```
{
  "forename": ".....",
  "surname": ".....",
  "email": ".....",
  "password": ".....",
  "isCustomer": True
}
```

Sva polja su obavezna i njihov sadržaj je sledeći:

- **forename**: string od najviše 256 karaktera koji predstavlja ime korisnika;
- **surname**: string od najviše 256 karaktera koji predstavlja prezime korisnika;
- **email**: string od najviše 256 karaktera koji predstavlja email adresu korisnika;
- **password**: string od najviše 256 karaktera koji predstavlja lozinku korisnika, dužina lozinke mora biti 8 ili više znakova, lozinka mora sadržati bar jednu cifru, jednom malo slovo i jedno veliko slovo;
- **isCustomer**: logička vrednost, ukoliko je **True** podaci se odnose na kupca, ukoliko je **False** podaci se odnose na magacionera;

Odgovor Ukoliko su svi traženi podaci prisutni u telu zahteva i ispunjavaju navedena ograničenja, rezultat zahteva je kreiranje novog korisnika sa ulogom kupca ili magacionera i odgovor sa statusnim kodom 200 bez dodatno sadržaja.

U slučaju greške, rezultat zahteva je odgovor sa statusnim kodom 400 čiji je sadržaj JSON objekat sledećeg formata:

```
{
    "message": "....."
}
```

Sadržaj polja **message** je:

- **"Field <fieldname> is missing."** ukoliko neko od polja nije prisutno ili je vrednost polja string dužine 0, <fieldname> je ime polja koje je očekivano u telo zahteva;
- **"Invalid email."** ukoliko polje **email** nije odgovarajuće formata;
- **"Invalid password."** ukoliko polje **password** nije odgovarajućeg formata;
- **"Email already exists."** ukoliko u bazi postoji korisnik sa istom email adresom;

Odgovarajuće provere se vrše u navedenom redosledu.

- **Prijava korisnika**

Adresa /login

Tip POST

Zaglavlja -

Telo Telo zahteva je JSON objekat sledećeg formata:

```
{
    "email": ".....",
    "password": "....."
}
```

Sva polja su obavezna i njihov sadržaj je sledeći:

- **email:** string od najviše 256 karaktera koji predstavlja email adresu korisnika;
- **password:** string od najviše 256 karaktera koji predstavlja lozinku korisnika;

Odgovor Ukoliko su svi traženi podaci prisutni u telu zahteva i ispunjavaju navedena ograničenja i u bazi podatak postoji korisnik sa navedenim kredencijalima, rezultat zahteva je odgovor sa statusnim kodom 200 čiji je sadržaj JSON objekat sledećeg formata:

```
{
    "accessToken": ".....",
    "refreshToken": "....."
}
```

Sadržaj polja je sledeći:

- **accessToken:** string koji predstavlja JSON veb token koji se koristi za pristup ostalim funkcionalnostima sistema, token je validan narednih sat vremena;
- **refreshToken:** string koji predstavlja JSON veb token koji se koristi za ponovno izdavanje novog JSON veb token za pristup, token je validan narednih 30 dana;

Lice za koje se token izdaje se identifikuje na osnovu email adrese. Oba tokena imaju dodatna polja čiji sadržaj predstavlja informacije koje je korisnik zadao prilikom registracije. Imena polja su ista kao ona navedena u opisu registracije.

U slučaju greške, rezultat zahteva je odgovor sa statusnim kodom 400 i JSON objektom sledećeg formata:

```
{
    "message": "....."
}
```

Sadržaj polja **message** je:

- **"Field <fieldname> is missing."** ukoliko neko od polja nije prisutno ili je vrednost polja string dužine 0, **<fieldname>** je ime polja koje je očekivano u telo zahteva;
- **"Invalid email."** ukoliko polje email nije odgovarajućeg formata;
- **"Invalid credentials."** ukoliko korisnik ne postoji;

Odgovarajuće provere se vrše u navedenom redosledu.

- **Osvežavanje tokena za pristup**

Adresa /refresh

Tip POST

Zaglavlja Zaglavlja i njihov sadržaj su:

```
{
    "Authorization": "Bearer <REFRESH_TOKEN>"
}
```

Vrednost **<REFRESH_TOKEN>** je string koji predstavlja JSON veb token za osvežavanje.

Telo -

Odgovor Ukoliko su sva zaglavlja prisutna i ispunjavaju navedena ograničenja, rezultat zahteva je odgovor sa statusnim kodom 200 čiji je sadržaj JSON objekat sledećeg formata:

```
{
    "accessToken": "....."
}
```

Sadržaj polja **accessToken** je string koji predstavlja JSON veb token koji se koristi za pristup ostalim funkcionalnostima sistema. Token je validan narednih sat vremena i njegov sadržaj je opisan u odeljku koji se odnosi na prijavu.

U slučaju da je zaglavlje izostavljeno, rezultat je odgovor sa statusnim kodom 401 i JSON objektom sledećeg formata i sadržaja:

```
{
    "msg": "Missing Authorization Header"
}
```

- **Brisanje korisnika**

Adresa /delete

Tip POST

Zaglavlja Zaglavlja i njihov sadržaj su:

```
{
    "Authorization": "Bearer <ACCESS_TOKEN>"
}
```

Vrednost **<ACCESS_TOKEN>** je string koji predstavlja JSON veb token za pristup koji je izdat administratoru prilikom prijave.

Telo

Telo zahteva je JSON objekat sledećeg formata:

```
{  
    "email": "....."  
}
```

Polje **email** je obavezno. Sadržaj polja je string od najviše 256 karaktera koji predstavlja email adresu korisnika.

Odgovor

Ukoliko su sva tražena zaglavlja prisutna i svi traženi podaci prisutni u telu zahteva i ispunjavaju navedena ograničenja, rezultat zahteva je brisanje korisnika iz baze podataka i odgovor sa statusnim kodom 200 bez dodatnog sadržaja.

U slučaju da zaglavlje nedostaje, rezultat je odgovor sa statusnim kodom 401 i JSON objektom sledećeg formata i sadržaja:

```
{  
    "msg": "Missing Authorization Header"  
}
```

U slučaju greške, rezultat zahteva je odgovor sa statusnim kodom 400 i JSON objektom sledećeg formata:

```
{  
    "message": "....."  
}
```

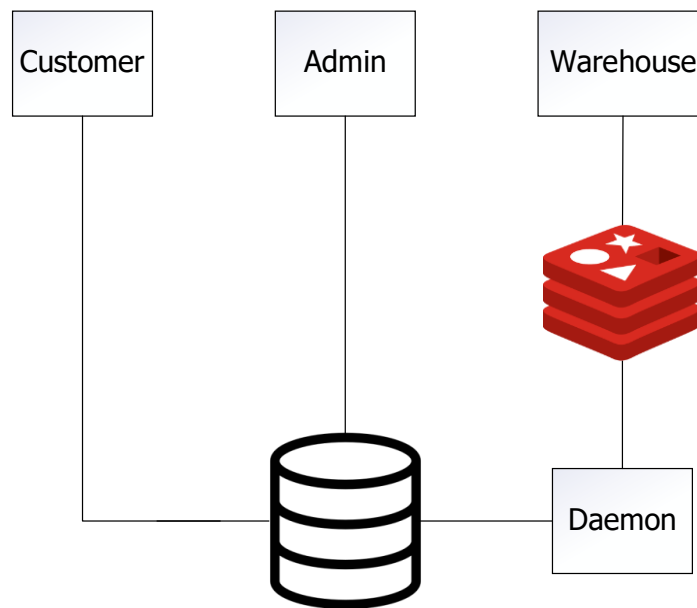
Sadržaj polja **message** je:

- **"Field email is missing."** ukoliko polje **email** nije prisutno ili je vrednost polja string dužine 0;
- **"Invalid email."** ukoliko polje **email** nije odgovarajućeg formata;
- **"Unknown user."** ukoliko korisnik sa datom email adresom ne postoji;

Odgovarajuće provere se vrše u navedenom redosledu.

Upravljanje prodavnicom

Izgled dela sistema koji je zadužen za upravljanje prodavnicom dat je slici 2.



Slika 2. Izgled dela sistema za upravljanje prodavnicom

Ovaj deo sistema se sastoji iz baze podatak u kojoj se čuvaju sve informacije neophodne za realizaciju ovog dela sistema, **Redis** veb servisa, kontejnera koji predstavlja veb servis sa funkcionalnostima dostupni administratorima, kontejnera koji predstavlja veb servis sa funkcionalnostima dostupnim kupcima, kontejnera koji predstavlja veb servis sa funkcionalnostima dostupnim magacionerima i kontejnera u kojem se izvršava proces koji je zadužen za proveru pristiglih proizvoda. U nastavku su dati opisi funkcionalnosti svakog kontejnera. Svaka adresa je relativna u odnosu na IP adresu kontejnera i broj porta na kojem kontejner sluša. Svaka od funkcionalnosti zahteva odgovarajući token za pristup.

Funkcionalnosti kontejnera koji namenjen za rad sa magacionerima su date u nastavku.

- **Dostava proizvoda**

Adresa /update

Tip POST

Zaglavlja Zaglavlja i njihov sadržaj su:

```
{
  "Authorization": "Bearer <ACCESS_TOKEN>"
}
```

Vrednost **<ACCESS_TOKEN>** je string koji predstavlja JSON veb token za pristup koji je izdat magacioneru prilikom prijave.

Telo U telu zahteva se nalazi polje sa nazivom **file** čija vrednost predstavlja CSV datoteku sa informacijama o pristiglim proizvodima. Svaki red datoteke odgovara jednom proizvodu i sadrži sledeće vrednosti:

- Imena kategorija kojima ovaj proizvod pripada, imena su razdvojena znakom "|";
- Ime proizvoda;
- Količina (ceo broj);
- Nabavna cena (realan broj);

Odgovor Ukoliko su svi podaci prisutni i ispunjavaju sva navedena ograničenja, informacije o proizvodima se smeštaju na **Redis** servis i rezultat zahteva je odgovor sa statusnim kodom 200 bez dodatnog sadržaja.

U slučaju da je zaglavlje izostavljeno, rezultat je odgovor sa statusnim kodom 401 i JSON objektom sledećeg formata i sadržaja:

```
{
    "msg": "Missing Authorization Header"
}
```

U slučaju da neko polje tela nedostaje, rezultat zahteva je odgovor sa statusnim kodom 400 i JSON objektom sledećeg formata:

```
{
    "message": "....."
}
```

Sadržaj polja **message** je:

- **"Field file missing."** ukoliko polje **file** nije prisutno u telu zahteva;
- **"Incorrect number of values on line 2."** ukoliko neka od linija u datoteci ne sadrži tačno četiri vrednost, poruka treba da sadrži i broj linije koja ne zadovoljava uslov, numeracija linija kreće od 0;
- **"Incorrect quantity on line 2."** ukoliko neka od linija u datoteci ne sadrži odgovarajući količinu, količina je ceo broj veći od 0, poruka treba da sadrži i broj linije koja ne zadovoljava uslov, numeracija linija kreće od 0;
- **"Incorrect price on line 2."** ukoliko neka od linija u datoteci ne sadrži odgovarajući nabavnu cenu, nabavna cena je realan broj veći od 0, poruka treba da sadrži i broj linije koja ne zadovoljava uslov, numeracija linija kreće od 0;

Odgovarajuće provere se vrše u navedenom redosledu.

U kontejneru sa nazivom **Dameon** se izvršava proces čije je zaduženje provera validnosti podataka o pristiglim proizvodima. Proces u ovom kontejneru čeka da se informacije o pristiglim proizvodima upišu na **Redis** servis nakon čega ih dohvata i vrši odgovarajuće provere. Proces najpre proverava da li u bazi postoji proizvod sa istim imenom. Ukoliko ne postoji, proces dodaje proizvod u bazu podataka zajedno sa svim njegovim kategorijama. Ukoliko neka kategorija nije prisutna u bazi podataka, potrebno ju je napraviti. Ukoliko se proizvod nalazi u bazi podataka, onda je potrebno proveriti da li je lista kategorija navedena prilikom preuzimanja proizvoda ista kao ona u bazi podataka. Ukoliko nije, informacije o proizvodu se odbacuju kao nekorektne. Ukoliko jeste, potrebno je ažurirati količinu i cenu proizvoda. Cena proizvoda se ažurira na osnovu sledeće formule

$$newPrice = \frac{(currentQuantity * currentPrice + deliveredQuantity * deliveryPrice)}{currentQuantity + deliveryQuantity}$$

Značenje simbola navedenih u formuli je sledeće:

- **newPrice** – nova cena datog proizvoda;
- **currentQuantity** – trenutna količina;
- **currentPrice** – trenutna cena proizvoda;
- **deliveryQuantity** – dostavljena količina;
- **deliveryPrice** – nabavna cena;

Nakon ažuriranja podataka o proizvodu, ovaj proces proverava da li u bazi podataka postoje narudžbine na čekanju kojima nedostaje trenutno dodati proizvod. Ukoliko postoje, potrebno je da ovaj proces ažurira date narudžbine po redosledu kreiranja (FIFO pristup).

Funkcionalnosti kontejnera koji namenjen za rad sa kupcima su date u nastavku.

- **Pretraga proizvoda**

Adresa /search?name=<PRODUCT_NAME>&category=<CATEGORY_NAME>

Parametri <PRODUCT_NAME> i <CATEGORY_NAME> su stringovi koji predstavljaju parametre za pretragu proizvoda. Oba parametra su opcionalna.

Tip GET

Zaglavlja Zaglavlja i njihov sadržaj su:

```
{
    "Authorization": "Bearer <ACCESS_TOKEN>"
}
```

Vrednost <ACCESS_TOKEN> je string koji predstavlja JSON veb token za pristup koji je izdat kupcu prilikom prijave.

Telo -

Odgovor Ukoliko su sva tražena zaglavlja prisutna, rezultat zahteva je odgovor sa statusnim kodom 200 čiji je sadržaj JSON objekat sledećeg formata:

```
{
  "categories": [
    "Category2",
    "Category1",
    ...
  ],
  "products": [
    {
      "categories": [
        "Category1",
        "Category2",
        ...
      ],
      "id": 3,
      "name": "Product2",
      "price": 29.89,
      "quantity": 5
    },
    ...
  ]
}
```

Sadržaj polja **categories** je lista stringova. Svaki string predstavlja kategoriju u čijem se imenu nalazi string koji je prosleđen kao parametar **category** i kojoj pripada makar jedan proizvod u čijem se imenu nalazi parametar **name**.

Sadržaj polja **products** je lista JSON objekata. Svaki objekat predstavlja ime proizvoda u čijem se imenu nalazi string koji je prosleđen kao parametar **name** i koji pripada makar jednoj kategoriji u čijem se imenu nalazi parametar **category**.

Svaki od JSON objekata u listi **products** sadrži sledeća polja:

- **categories** – niz stringova koji predstavljaju imena kategorija kojima ovaj proizvod pripada;
- **id** – celobrojni identifikator proizvoda;
- **name** – string koji predstavlja ime proizvoda;
- **price** – realan broj koji predstavlja trenutnu cenu proizvoda;
- **quantity** – ceo broj koji predstavlja količinu;

U slučaju da zaglavlje nedostaje, rezultat je odgovor sa statusnim kodom 401 i JSON objektom sledećeg formata i sadržaja:

```
{
  "msg": "Missing Authorization Header"
}
```

- **Naručivanje**

Adresa /order

Tip POST

Zaglavlja Zaglavlja i njihov sadržaj su:

```
{
  "Authorization": "Bearer <ACCESS_TOKEN>"
}
```

Vrednost **<ACCESS_TOKEN>** je string koji predstavlja JSON veb token za pristup koji je izdat kupcu prilikom prijave.

Telo Telo zahteva je JSON objekat sledećeg formata:

```
{
  "requests": [
    {
      "id": 1,
      "quantity": 2
    },
    {
      "id": 2,
      "quantity": 3
    },
    ...
  ]
}
```

Sva polja su obavezna. Polje **requests** predstavlja niz proizvoda koje kupac želi da kupi. Svaki element predstavlja JSON objekat koji sadrži dva polje. Polje **id** predstavlja identifikator proizvoda, a polje **quantity** predstavlja količinu koju kupac želi da kupi.

Odgovor Ukoliko su sva tražena zaglavlja prisutna i svi traženi podaci prisutni u telu zahteva i ispunjavaju navedena ograničenja, rezultat zahteva je kreiranje narudžbine i odgovor sa statusnim kodom 200 čiji je sadržaj JSON objekat sledećeg formata:

```
{
  "id": 1
}
```

Polje **id** predstavlja celobrojni identifikator kreirane narudžbine. Prilikom kreiranja narudžbine potrebno je ispuniti sve zahteve kupca ukoliko je to moguće. Za sve neispunjene zahteve potrebno je sačekati novu dostavu proizvoda. Prilikom kreiranja narudžbine za cenu proizvoda se uzima trenutna cena i nove dostave proizvoda neće uticati na cenu narudžbine.

U slučaju da zaglavlje nedostaje, rezultat je odgovor sa statusnim kodom 401 i JSON objektom sledećeg formata i sadržaja:

```
{
  "msg": "Missing Authorization Header"
}
```

U slučaju da neko polje tela nedostaje ili je nekorektno, rezultat zahteva je odgovor sa statusnim kodom 400 i JSON objektom sledećeg formata:

```
{  
    "message": "....."  
}
```

Sadržaj polja **message** je:

- "Field requests is missing." ukoliko polje **requests** nije prisutno;
- "Product id is missing for request number 1." ukoliko u nekom od JSON objekata nedostaje polje **id**, numeracija objekata kreće od 0;
- "Product quantity is missing for request number 1." ukoliko u nekom od JSON objekata nedostaje polje **quantity**, numeracija objekata kreće od 0;
- "Invalid product id for request number 1." ukoliko neki od identifikatora proizvoda nije ceo broj veći od 0, numeracija objekata kreće od 0;
- "Invalid product quantity for request number 1." ukoliko neka od traženih količina nije ceo broj veći od 0, numeracija objekata kreće od 0;
- "Invalid product for request number 1." ukoliko neki od navedenih proizvoda ne postoji u bazi podataka, numeracija objekata kreće od 0;

Odgovarajuće provere se vrše u navedenom redosledu.

- **Pregled narudžbina**

Adresa /status

Tip GET

Zaglavlja Zaglavlja i njihov sadržaj su:

```
{  
    "Authorization": "Bearer <ACCESS_TOKEN>"  
}
```

Vrednost **<ACCESS_TOKEN>** je string koji predstavlja JSON veb token za pristup koji je izdat kupcu prilikom prijave.

Telo -

Odgovor

Ukoliko su sva tražena zaglavlja prisutna i ispunjavaju navedena ograničenja, rezultat zahteva je odgovor sa statusnim kodom 200 čiji je sadržaj JSON objekat sledećeg formata:

```
{
  "orders": [
    {
      "products": [
        {
          "categories": [
            "Category0",
            ...
          ],
          "name": "Product0",
          "price": 27.34,
          "received": 2,
          "requested": 2
        },
        ...
      ],
      "price": 179.0,
      "status": "COMPLETE",
      "timestamp": "2022-05-22T20:32:17Z"
    },
    ...
  ]
}
```

Polje **orders** predstavlja niz JSON objekat gde svaki objekat predstavlja jednu narudžbinu kupca koji je pozvao ovaj servis. Svaki objekat sadrži sledeća polja:

- **price** – ukupna cena narudžbine;
- **status** – status narudžbine, **COMPLETE** ukoliko je kompletirana ili **PENDING** ukoliko je neophodno sačekati dostavu nekih proizvoda;
- **timestamp** – string koji predstavlja datum i vreme kreiranja narudžbine dat u [ISO 8601](#) formatu;
- **orders** – niz JSON objekata u kojem svaki objekat predstavlja jedan proizvodi sadrži sledeća polja:
 - **categories** – niz stringova koji predstavljaju imena kategorija kojima ovaj proizvod pripada;
 - **id** – celobrojni identifikator proizvoda;
 - **name** – string koji predstavlja ime proizvoda;
 - **price** – realan broj koji predstavlja cenu proizvoda u trenutku naručivanja;
 - **requested** – ceo broj koji predstavlja traženu količinu;
 - **received** – ceo broj koji predstavlja ispunjenu količinu

U slučaju da zaglavlje nedostaje, rezultat je odgovor sa statusnim kodom 401 i JSON objektom sledećeg formata i sadržaja:

```
{
  "msg": "Missing Authorization Header"
}
```

Funkcionalnosti kontejnera koji namenjen za rad sa administratorima su date u nastavku.

- **Dohvatanje statistike o proizvodima**

Adresa /productStatistics

Tip GET

Zaglavlja Zaglavlja i njihov sadržaj su:

```
{  
    "Authorization": "Bearer <ACCESS_TOKEN>"  
}
```

Vrednost **<ACCESS_TOKEN>** je string koji predstavlja JSON veb token za pristup koji je izdat administratoru prilikom prijave.

Telo -

Odgovor Ukoliko su sva tražena zaglavlja prisutna i ispunjavaju navedena ograničenja, rezultat zahteva je odgovor sa statusnim kodom 200 čiji je sadržaj JSON objekat sledećeg formata:

```
{  
    "statistics": [  
        {  
            "name": "Product0",  
            "sold": 4,  
            "waiting": 1  
        },  
        {  
            "name": "Product1",  
            "sold": 12,  
            "waiting": 2  
        },  
        {  
            "name": "Product9",  
            "sold": 4,  
            "waiting": 0  
        },  
        ...  
    ]  
}
```

Polje **statistics** predstavlja niz JSON objekata. Svaki objekat predstavlja proizvod koji ima makar jednu prodaju i sadrži sledeća polja:

- **name** – ime proizvoda;
- **sold** – ukupno količina prodatih proizvoda, uzeti u obzir i narudžbine koje nisu kompletne;
- **waiting** – broj proizvoda koji treba dostaviti da bi se potražnja za ovim proizvodom kompletno ispunila;

U slučaju da zaglavlje nedostaje, rezultat je odgovor sa statusnim kodom 401 i JSON objektom sledećeg formata i sadržaja:

```
{  
    "msg": "Missing Authorization Header"  
}
```

• Dohvatanje statistike o kategorijama

Adresa /categoryStatistics

Tip GET

Zaglavlja Zaglavlja i njihov sadržaj su:

```
{  
    "Authorization": "Bearer <ACCESS_TOKEN>"  
}
```

Vrednost **<ACCESS_TOKEN>** je string koji predstavlja JSON veb token za pristup koji je izdat administratoru prilikom prijave.

Telo

-

Odgovor

Ukoliko su sva tražena zaglavlja prisutna i ispunjavaju navedena ograničenja, rezultat zahteva je odgovor sa statusnim kodom 200 čiji je sadržaj JSON objekat sledećeg formata:

```
{
  "statistics": [
    "Category1",
    "Category0",
    "Category2",
    "Category3",
    "Category4",
    "Category5",
    "Category6",
    ...
  ]
}
```

Polje **statistics** predstavlja niz imena svih kategorija koje se trenutno nalaze u bazi. Niz je sortiran opadajuće po broju prodatih proizvoda koji pripadaju toj kategoriji. Ukoliko se proizvod pripada više od jednoj kategoriji, računati ga kao prodaju u svim njegovim kategorijama. Ukoliko dve kategorije imaju isti broj prodaja, sortirati ih rastuće po imenu.

U slučaju da zaglavlje nedostaje, rezultat je odgovor sa statusnim kodom 401 i JSON objektom sledećeg formata i sadržaja:

```
{
  "msg": "Missing Authorization Header"
}
```

Pokretanje Sistema

Potrebno je napisati konfiguracioni fajl pomoću kojeg se sistem može pokrenuti na klasteru računara. Konfiguracioni fajl je namenjen za **Docker** alat i klaster računara povezanih u **Docker Swarm**. Potrebno je obezbediti sledeća ograničenja:

- Potrebno je zabraniti spoljašnji pristup bazama podataka i obezbediti njihovu automatsku inicijalizaciju prilikom pokretanja celog sistema, inicijalizacija podrazumeva kreiranje svih neophodnih tabela i dodavanje jednog korisnika koji ima ulogu administratora sa sledećim podacima:

```
{
  "forename": "admin",
  "surname": "admin",
  "email": "admin@admin.com",
  "password": "1"
}
```

- Potrebno je obezbediti očuvanja podataka u bazama podataka ukoliko odgovarajući kontejneri prestanu sa radom;
- Potrebno je obezbediti da kontejneri iz drugog dela sistema ne mogu da pristupe kontejneru koji obezbeđuje autentikaciju i autorizaciju i bazi podataka u kojoj se nalaze korisnički nalozi. Pored toga, obezbediti da ova dva kontejnera mogu da komuniciraju sa ostatkom sistema;
- Potrebno je obezbediti da kontejner koji je namenjen za rad sa magacionerima može da pristupi samo **Redis** servisu;
- Potrebno je obezbediti da kontejner u kojem se izvršava **daemon** proces može da pristupi samo **Redis** servisu i bazi podataka u kojoj se nalaze podaci o prodavnicima;

- Potrebno je obezbediti da kontejner koji je namenjen za rad sa administratorima može da pristupi samo bazi podataka u kojoj se nalaze podaci o prodavnicima;
- Potrebno je obezbediti da kontejner koji je namenjen za rad sa kupcima može da pristupi samo bazi podataka u kojoj se nalaze podaci o prodavnicima;
- Potrebno je pokrenuti 3 replike kontejnera koji je namenjen za rad sa magacionerima;
- Potrebno je pokrenuti 3 replike kontejnera koji je namenjen za rad sa kupcima;

Na samoj odbrani, od studenata se očekuje da pomoću ovog fajla pokrenu sistem i demonstriraju njegov rad.