

# **Систем за управљање рада продавником**

# Увод

Основни циљеви пројекта су:

- имплементација веб сервиса који чине дати систем;
- покретање система помоћу алата за оркестрирање контејнера;

Систем је намењен за вишекориснички рад и стога треба пажљиво да буде дизајниран за потребе исправног и ефикасног рада. Део функционалности система је јавно доступан, док је део функционалности обезбеђен само за кориснике који могу да се пријаве на систем.

Систем је потребно имплементирати коришћењем **Python** програмског језика, **Flask** рандог оквира и **SQLAlchemy** библиотека. Приликом обраде захтева корисника неопходно је користити **SQLAlchemy** у што већој мери, односно реализовати обраду помоћу упита свугде где је то могуће. Поред кода, потребно је приложити датотеке на основу којих се креирају **Docker Image** артефакти који представљају делове система и који се могу искористити за покретање одговарајућих контејнера. Поред имплементације, потребно је написати конфигурациони фајл помоћу којег се цео систем може покренути коришћењем **Docker** алата.

## Концептуални опис система

Систем треба да обезбеди регистрацију корисника (купац или курир). Купци могу да врше претрагу и да праве наруџбине. Курири могу да врше достављање наруџбина.

Приликом покретања система потребно је унапред обезбедити налоге свим власницима продавнице. Власници продавнице могу да додају производе и да врше преглед статистике продаја.

Сваки корисник треба да буде регистрован пре коришћења система. За сваког корисника се у оквиру његовог корисничког налога чувају следеће информације: имејл адреса и лозинка које се користе приликом пријаве, име, презиме и улога корисника. Корисник може имати улогу купца, власника продавнице или курира.

За производе се памте следеће информације: категорија којој производ припада (може их бити више), име производа и његова цена. Име производа је јединствено. За сваку категорију се памти њено име.

Купци могу да врше претрагу производа, да праве наруџбине и да врше преглед направљених наруџбина.

Приликом сваке куповине производа у систему се прави наруџбина. За сваку наруџбину се чува списак производа, укупна цена наруџбине, њен статус и тренутак њеног креирања. Приликом креирања, наруџбина је у стању „чекања“ све док је неки од курира не покупи када прелази у стање „на путу“. Када курир достави наруџбину купцу она прелази у стање „извршена“.

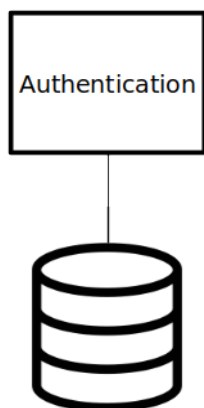
## Функционални опис система

Систем се састоји из два дела: један намењен за управљање корисничким налозима и један намењен за управљање продавницом. И један и други део система се састоје из неколико компоненти које су релизоване помоћу

контејнера креираних на основу **Docker Image** шаблона. Део ових **Docker Image** шаблона већ постоји и налази се у оквиру јавног репозиторијума на адреси <https://hub.docker.com/>. Остатак је неопходно имплементирати. У наставку је дат функционални опис система.

## Управљање корисничким налозима

Изглед дела система који је задужен за управљање корисничким налозима дат је на слици 1.



Слика 1. Изглед дела система за управљање корисничким налозима

Овај део система се састоји из једног контејнера и базе података у којој се налазе само подаци о корисницима. Контејнер представља веб сервис помоћу којег корисник може да се региструје и добије [JSON веб токен](#) са којим ће моћи да приступи остатку система.

Опис функционалности које пружа овај веб сервис дат је у наставку. Свака адреса је релативна у односу на ИП адресу контејнера и број порта на којем контејнер слуша.

- **Регистрација корисника**

Адресе      /register\_customer

              /register\_courier

Тип        POST

Заглавље   -  
е

Тело        Тело захтева је JSON објекат следећег формата:

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

Сва поља су обавезна и њихов садржај је следећи:

- **forename**: стринг од највише 256 карактера који представља име корисника;
- **surname**: стринг од највише 256 карактера који представља презиме корисника;
- **email**: стринг од највише 256 карактера који представља имејл адресу корисника;

- **password:** стринг од највише 256 карактера који представља лозинку корисника, дужина лозинке мора бити 8 или више знакова;

Одговор Уколико су сви тражени подаци присутни у телу захтева и испуњавају наведена ограничења, резултат захтева је креирање новог корисника са улогом купца или курира (у зависности од адресе на коју је послат захтев) и одговор са статусним кодом 200 без додатно садржаја.

У случају грешке, резултат захтева је одговор са статусним кодом 400 чији је садржај JSON објекат следећег формата:

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

Садржај поља **message** је:

- **"Field <fieldname> is missing."** уколико неко од поља није присутно или је вредност поља стринг дужине 0, **<fieldname>** је име поља које је очекивано у тело захтева;
- **"Invalid email."** уколико поље **email** није одговарајуће формата;
- **"Invalid password."** уколико поље **password** није одговарајуће дужине;
- **"Email already exists."** уколико у бази постоји корисник са истом имејл адресом;

Одговарајуће провере се врше у наведеном редоследу.

## • Пријава корисника

Адреса /login

Тип POST

Заглавље -

е

Тело Тело захтева је JSON објекат следећег формата:

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

Сва поља су обавезна и њихов садржај је следећи:

- **email:** стринг од највише 256 карактера који представља имејл адресу корисника;
- **password:** стринг од највише 256 карактера који представља лозинку корисника;

Одговор Уколико су сви тражени подаци присутни у телу захтева и испуњавају наведена ограничења и у бази податак постоји корисник са наведеним креденцијалима, резултат захтева је одговор са статусним кодом 200 чији је садржај JSON објекат следећег формата:

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

Поље **accessToken** је стринг који представља JSON веб токен који се користи за приступ осталим функционалностима система. Токен је валидан наредних сат времена. Лице за које се токен издаје се идентификује на основу имејл адресе. Токен садржи додатна поља чији садржај представља информације које је корисник задао приликом

регистрације. Имена поља су иста као она наведена у опису регистрације. У случају грешке, резултат захтева је одговор са статусним кодом 400 и JSON објектом следећег формата:

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

Садржај поља **message** је:

- **"Field <fieldname> is missing."** уколико неко од поља није присутно или је вредност поља стринг дужине 0, **<fieldname>** је име поља које је очекивано у тело захтева;
- **"Invalid email."** уколико поље имејл није одговарајућег формата;
- **"Invalid credentials."** уколико корисник не постоји;

Одговарајуће провере се врше у наведеном редоследу.

#### • Брисање корисника

Адреса /delete

Тип POST

Заглавље Заглавље и њихов садржај су:  
е

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

Вредност **<ACCESS\_TOKEN>** је стринг који представља JSON веб токен за приступ који је издат кориснику који жели да обрише свој налог.

Тело -

Одговор Уколико су сва тражена заглавља присутна, резултат захтева је брисање корисника из базе података и одговор са статусним кодом 200 без додатног садржаја.

У случају да заглавље недостаје, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

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

У случају грешке, резултат захтева је одговор са статусним кодом 400 и JSON објектом следећег формата:

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

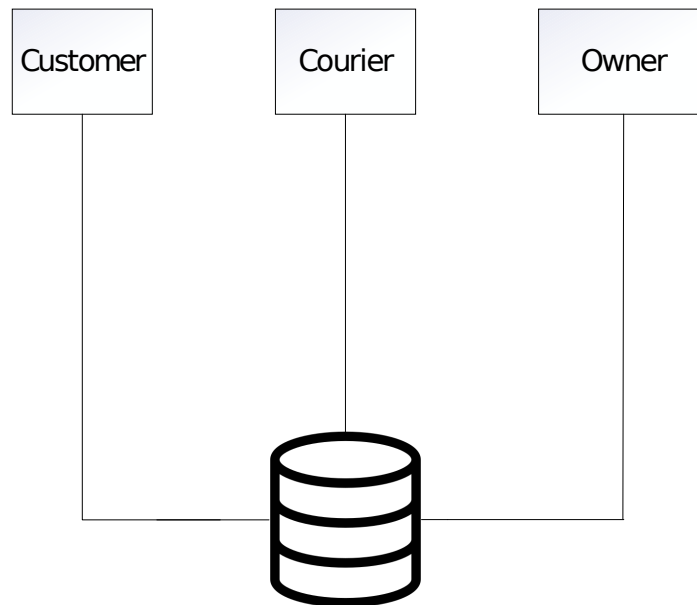
Садржај поља **message** је:

- **"Unknown user."** уколико корисник са датом имејл адресом не постоји;

Одговарајуће провере се врше у наведеном редоследу.

# Управљање продавницом

Изглед дела система који је задужен за управљање продавницом дат је слици 2.



Слика 2. Изглед дела система за управљање продавницом

Овај део система се састоји из базе податак у којој се чувају све информације неопходне за рад продавнице, контејнера који представља веб сервис са функционалностима доступни власницима, контејнера који представља веб сервис са функционалностима доступним купцима и контејнера који представља веб сервис са функционалностима доступним куририма. У наставку су дати описи функционалности сваког контејнера. Свака адреса је релативна у односу на IP адресу контејнера и број порта на којем контејнер слуша. Свака од функционалности захтева одговарајући токен за приступ.

Функционалности контејнера који намењен за рад са власником су дате у наставку.

- **Додавање производа**

Адреса /update

Тип POST

Заглавље Заглавље и њихов садржај су:

е

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

Вредност **<ACCESS\_TOKEN>** је стринг који представља JSON веб токен за приступ који је издат власнику приликом пријаве.

Тело У телу захтева се налази поље са називом **file** чија вредност представља CSV датотеку са информацијама о производима које је потребно додати. Сваки ред датотеке одговара једном производу и садржи следеће вредности:

- Имена категорија којима овај производ припада, имена су раздвојена знаком “|”;
- Име производа;
- Нjegова цена (цео број);

Одговор Уколико су сви подаци присутни и испуњавају сва наведена ограничења, информације о производима се смештају у базу података и резултат захтева је одговор са статусним кодом 200 без додатног садржаја.

У случају да је заглавље изостављено, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

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

У случају да неко поље тела недостаје или је некоректног формата, резултат захтева је одговор са статусним кодом 400 и JSON објектом следећег формата:

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

Садржај поља `message` је:

- **"Field file missing."** уколико поље `file` није присутно у телу захтева;
- **"Incorrect number of values on line 2."** уколико нека од линија у датотеци не садржи тачно три вредност, порука треба да садржи и број линије која не задовољава услов, нумерација линија креће од 0;
- **"Incorrect price on line 2."** уколико нека од линија у датотеци не садржи одговарајући цену, цена је реалан број већи од 0, порука треба да садржи и број линије која не задовољава услов, нумерација линија креће од 0;
- **"Product <NAME> already exists."** уколико производ са називом `<NAME>` већ постоји, уколико макар један од производа у прослеђеној датотеци већ постоји не треба додати ниједан производ из прослеђене датотеке;

Одговарајуће провере се врше у наведеном редоследу.

- **Дохватање статистике о производима**

Адреса /product\_statistics

Тип GET

Заглавље Заглавље и њихов садржај су:  
е

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

Вредност `<ACCESS_TOKEN>` је стринг који представља JSON веб токен за приступ који је издат власнику приликом пријаве.

Тело

-

Одговор Уколико су сва тражена заглавља присутна и испуњавају наведена ограничења, резултат захтева је одговор са статусним кодом 200 чији је садржај JSON објекат следећег формата:

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

Поље **statistics** представља низ JSON објеката. Сваки објекат представља производ који има макар једну продају и садржи следећа поља:

- **name** – име производа;
- **sold** – укупно количина продатих примерака датог производа, узети у обзир само оне производе који припадају наруџбинама које су достављене купцима;
- **waiting** – број примерака производа који припадају наруџбинама које тек треба доставити купцима;

У случају да заглавље недостаје, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

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

#### • Дохватање статистике о категоријама

Адреса /category\_statistics

Тип GET

Заглавље Заглавље и њихов садржај су:  
е

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

Вредност **<ACCESS\_TOKEN>** је стринг који представља JSON веб токен за приступ који је издат власнику приликом пријаве.

Тело -

Одговор Уколико су сва тражена заглавља присутна и испуњавају наведена ограничења, резултат захтева је одговор са статусним кодом 200 чији је садржај JSON објекат следећег формата:

```
{
  "statistics": [
    "Category1",
    "Category0",
    "Category2",
  ]
}
```



```

        "Category3",
        "Category4",
        "Category5",
        "Category6",
        ...
    ]
}

```

Поље **statistics** представља низ имена свих категорија које се тренутно налазе у бази. Низ је сортиран опадајуће по броју достављених примерака производа који припадају тој категорији. Уколико се производ припада више од једној категорији, рачунати га као достављеног у свим његовим категоријама. Уколико две категорије имају исти број достављених производа, сортирати их растуће по имену.

У случају да заглавље недостаје, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

```

{
    "msg": "Missing Authorization Header"
}

```

Функционалности контејнера који намењен за рад са купцима су дате у наставку.

- **Претрага производа**

Адреса      /search?name=<PRODUCT\_NAME>&category=<CATEGORY\_NAME>

Параметри <PRODUCT\_NAME> и <CATEGORY\_NAME> су стрингови који представљају параметре за претрагу производа. Оба параметра су опциона.

Тип            GET

Заглавље    Заглавље и њихов садржај су:

```

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

```

Вредност <ACCESS\_TOKEN> је стринг који представља JSON веб токен за приступ који је издат купцу приликом пријаве.

Тело            -

Одговор        Уколико су сва тражена заглавља присутна, резултат захтева је одговор са статусним кодом 200 чији је садржај JSON објекат следећег формата:

```

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

```

Садржај поља **categories** је листа стрингова. Сваки стринг представља

категорију у чијем се имену налази стринг који је прослеђен као параметар **category** и којој припада макар један производ у чијем се имену налази параметар **name**.

Садржај поља **products** је листа JSON објеката. Сваки објекат представља име производа у чијем се имену налази стринг који је прослеђен као параметар **name** и који припада макар једној категорији у чијем се имену налази параметар **category**.

Сваки од JSON објеката у листи **products** садржи следећа поља:

- **categories** – низ стрингова који представљају имена категорија којима овај производ припада;
- **id** – целобројни идентификатор производа;
- **name** – стринг који представља име производа;
- **price** – реалан број који представља тренутну цену производа;

У случају да заглавље недостаје, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

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

## • Наручивање

Адреса      /order

Тип            POST

Заглавље    Заглавље и њихов садржај су:  
е

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

Вредност **<ACCESS\_TOKEN>** је стринг који представља JSON веб токен за приступ који је издат купцу приликом пријаве.

Тело            Тело захтева је JSON објекат следећег формата:

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

Сва поља су обавезна. Поље **requests** представља низ производа које купац жели да купи. Сваки елемент представља JSON објекат који садржи два поља. Поље **id** представља идентификатор производа, а поље **quantity** представља количину коју купац жели да купи.

Одговор        Уколико су сва тражена заглавља присутна и сви тражени подаци присутни у телу захтева и испуњавају наведена ограничења, резултат захтева је креирање наруџбине и одговор са статусним кодом 200 чији је садржај JSON објекат следећег формата:

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

```
}
```

Поље **id** представља целобројни идентификатор креиране наруџбине.

У случају да заглавље недостаје, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

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

У случају да неко поље тела недостаје или је некоректно, резултат захтева је одговор са статусним кодом 400 и JSON објектом следећег формата:

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

Садржај поља **message** је:

- **"Field requests is missing."** уколико поље **requests** није присутно;
- **"Product id is missing for request number 1."** уколико у неком од JSON објеката недостаје поље **id**, нумерација објеката креће од 0;
- **"Product quantity is missing for request number 1."** уколико у неком од JSON објеката недостаје поље **quantity**, нумерација објеката креће од 0;
- **"Invalid product id for request number 1."** уколико неки од идентификатора производа није цео број већи од 0, нумерација објеката креће од 0;
- **"Invalid product quantity for request number 1."** уколико нека од тражених количина није цео број број већи од 0, нумерација објеката креће од 0;
- **"Invalid product for request number 1."** уколико неки од наведених производа не постоји у бази података, нумерација објеката креће од 0;

Одговарајуће провере се врше у наведеном редоследу.

## • Преглед наруџбина

Адреса      /status

Тип          GET

Заглавље    Заглавље и њихов садржај су:  
е

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

Вредност **<ACCESS\_TOKEN>** је стринг који представља JSON веб токен за приступ који је издат купцу приликом пријаве.

Тело          -

Одговор Уколико су сва тражена заглавља присутна и испуњавају наведена ограничења, резултат захтева је одговор са статусним кодом 200 чији је садржај JSON објекат следећег формата:

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

Поље **orders** представља низ JSON објекат где сваки објекат представља једну наруџбину купца који је позвао овај сервис. Сваки објекат садржи следећа поља:

- **price** – укупна цена наруџбине;
- **status** – статус наруџбине, **CREATED** уколико је креирана, **PENDING** уколико је неко од курира купио дату наруџбину или **COMPLETE** уколико је наруџбина достављена;
- **timestamp** – стринг који представља датум и време креирања наруџбине дат у [ISO 8601](#) формату;
- **orders** – низ JSON објеката у којем сваки објекат представља један производи садржи следећа поља:
  - о **categories** – низ стрингова који представљају имена категорија којима овај производ припада;
  - о **id** – целобројни идентификатор производа;
  - о **name** – стринг који представља име производа;
  - о **price** – реалан број који представља цену производа;
  - о **quantity** – цео број који представља тражену количину;

У случају да заглавље недостаје, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

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

- **Потврда доставе**

Адреса /delivered

Тип POST

Заглавље Заглавље и њихов садржај су:

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

Вредност **<ACCESS\_TOKEN>** је стринг који представља JSON веб токен за приступ који је издат купцу приликом пријаве.

Тело Тело захтева је JSON објекат следећег формата:

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

Поље **id** представља идентификатор пристигле наруџбине.

Одговор Уколико су сва тражена заглавља и поља присутна и испуњавају наведена ограничења, резултат захтева је промена стања одговарајуће наруџбине и одговор са статусним кодом 200 без садржаја.

У случају да заглавље недостаје, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

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

У случају да неко поље недостаје или је некоректно, резултат захтева је одговор са статусним кодом 400 и JSON објектом следећег формата:

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

Садржај поља **message** је:

- **"Missing order id."** уколико поље **id** недостаје;
- **"Invalid order id."** уколико поље **id** не представља цео број већи од 0 или у бази не постоји наруџбина са датим идентификатором или дату наруџбину није преузео неко од курира;

Одговарајуће провере се врше у наведеном редоследу.

Функционалности контејнера који намењен за рад са куририма су дате у наставку.

- **Преглед наруџбина које треба доставити**

Адреса /orders\_to\_deliver

Тип GET

Заглавље Заглавље и њихов садржај су:

е

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

Вредност **<ACCESS\_TOKEN>** је стринг који представља JSON веб токен за приступ који је издат куриру приликом пријаве.

Тело -

Одговор Уколико су сва тражена заглавља присутна и испуњавају наведена ограничења, резултат захтева је одговор са статусним кодом 200 чији је садржај JSON објекат следећег формата:

```
{
  "orders": [
    {
      "id": 1,
      "email": "..."
    },
    {
      "id": 2,
      "email": "..."
    },
    ...
  ]
}
```

Поље **orders** представља низ наруџбина које нису преузете. Сваки елемент представља JSON објекат који садржи два поља. Поље **id** представља идентификатор наруџбине, а поље **email** представља имејл адресу купца коме треба доставити наруџбину.

У случају да заглавље недостаје, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

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

#### • Преузимање наруџбине

Адреса /pick\_up\_order

Тип POST

Заглавље Заглавље и њихов садржај су:  
е

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

Вредност **<ACCESS\_TOKEN>** је стринг који представља JSON веб токен за приступ који је издат куриру приликом пријаве.

Тело Тело захтева је JSON објекат следећег формата:

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

Поље **id** представља идентификатор наруџбине коју курир жели да преузме и достави купцу.

Одговор Уколико су сва тражена заглавља присутна и сви тражени подаци присутни у телу захтева и испуњавају наведена ограничења, резултат захтева је промена стања одговарајуће наруџбине одговор са статусним кодом 200 без икаквог садржаја.

У случају да заглавље недостаје, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

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

У случају да неко поље недостаје или је некоректно, резултат захтева је одговор са статусним кодом 400 и JSON објектом следећег формата:

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

Садржај поља `message` је:

- **"Missing order id."** уколико поље `id` недостаје;
- **"Invalid order id."** уколико поље `id` не представља цео број већи од 0, уколико бази не постоји наруџбина са датим идентификатором или је већ преузета или достављена;

Одговарајуће провере се врше у наведеном редоследу.

## Покретање система

Потребно је написати конфигурациони фајл помоћу којег се систем може покренути. Конфигурациони фајл је намењен за **Docker-Compose** алат. Све информације неопходне за рад контејнер (као на пример URL базе подата) проследити као варијабле окружења. Потребно је обезбедити следећа ограничења:

- Потребно је забранити спољашњи приступ базама података и обезбедити њихову аутоматску иницијализацију приликом покретања целог система, иницијализација подразумева креирање свих неопходних табела и додавање једног власника са следећим информацијама:

```
{
  "forename": "Scrooge",
  "surname": "McDuck",
  "email": "onlymoney@gmail.com",
  "password": "evenmoremoney"
}
```

- Потребно је обезбедити очувања података у базама података уколико одговарајући контејнери престану са радом;
- Потребно је обезбедити да контејнери из дела система намењеном за аутентикацију и ауторизацију не могу да приступе контејнерима из дела система који служе за управљање рада продавнице и обратно.

На самој одбрани, од студената се очекује да помоћу овог фајла покрену систем и демонстрирају његов рад.

## Наплата

Потребно је обезбедити наплату наруџбина путем **Ethereum Blockchain** платформе. За симулацију ове платформе искористи **Docker Image** шаблоне доступан на [Docker Hub \(https://hub.docker.com/r/trufflesuite/ganache-cli/\)](https://hub.docker.com/r/trufflesuite/ganache-cli/) репозиторијуму.

За сваку креирану наруџбину потребно је креирати један паметни уговор. Обезбедити да доставу (а и везивање курира за уговор) није могуће извршити све док купац не пребаци тражену количину криптовалута (изражену у **wei** апоенима) на сам уговор. Количина криптовалута коју треба наплатити (изражена у **wei** апоенима) се добија помоћу формуле `order_price * 100`. По потврди доставе од стране купца, 80% пребачених средстава дати власнику продавнице, а осталих 20% пребацили куриру.

За наведеног власника креирати пар кључева и обезбедити му довољно количину криптовалута тако да може да покрије трошкове уговора који одговарају наруџбинама. Средства пребацили са првог од десет рачуна који се креирају приликом покретања симулатора платформе.

Приликом креирања уговора потребно га је везати за купца који креирао наруџбину, али потребно оставити опцију да се курир тек касније прикључи уговору. Власник такође сноси трошкове везивања курира за неки од уговора.

Приликом пребацивања средстава од стране купца, потребно је обезбедити да само онај купац који је везан за дати уговор може да изврши пребацивања средстава. Поред тога, обезбедити да везани купац мора да пребаци тачну своту новца пре него што курир може да преузме наруџбину и крене са доставом. Купац сноси трошак пребацивања средстава.

Приликом одабира наруџбине за доставу од стране курира везати истог за одговарајући уговор. Приликом потврде доставе од стране купца извршити трансфер средстава на рачун курира и власника и забранити било коју даљу интеракцију са одговарајућим уговором. Забранити купцима да врше потврду доставе пре везивања неког курира за дати уговор.

Приликом реализације наплата потребно је дорадити следеће функционалности контејнера који намењен за рад са купцима (промене су обојене сивом бојом):

- **Наручивање**

Адреса      /order

Тип            POST

Заглавље    Заглавље и њихов садржај су:

е                {  
                  "Authorization": "Bearer <ACCESS\_TOKEN>"  
              }

Вредност <ACCESS\_TOKEN> је стринг који представља JSON веб токен за приступ који је издат купцу приликом пријаве.

Тело            Тело захтева је JSON објекат следећег формата:

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

Сва поља су обавезна. Поље **requests** представља низ производа које купац жели да купи. Сваки елемент представља JSON објекат који садржи два поља. Поље **id** представља идентификатор производа, а поље **quantity** представља количину коју купац жели да купи. Поље **address** је стринг који представља рачун датог купца.

Одговор        Уколико су сва тражена заглавља присутна и сви тражени подаци присутни у телу захтева и испуњавају наведена ограничења, резултат захтева је креирање наруџбине, креирање одговарајућег паметног уговора и одговор са статусним кодом 200 чији је садржај JSON објекат следећег формата:

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

Поље **id** представља целобројни идентификатор креиране наруџбине.

У случају да заглавље недостаје, резултат је одговор са статусним кодом



401 и JSON објектом следећег формата и садржаја:

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

У случају да неко поље тела недостаје или је некоректно, резултат захтева је одговор са статусним кодом 400 и JSON објектом следећег формата:

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

Садржај поља **message** је:

- **"Field requests is missing."** уколико поље **requests** није присутно;
- **"Product id is missing for request number 1."** уколико у неком од JSON објеката недостаје поље **id**, нумерација објеката креће од 0;
- **"Product quantity is missing for request number 1."** уколико у неком од JSON објеката недостаје поље **quantity**, нумерација објеката креће од 0;
- **"Invalid product id for request number 1."** уколико неки од идентификатора производа није цео број већи од 0, нумерација објеката креће од 0;
- **"Invalid product quantity for request number 1."** уколико нека од тражених количина није цео број број већи од 0, нумерација објеката креће од 0;
- **"Invalid product for request number 1."** уколико неки од наведених производа не постоји у бази података, нумерација објеката креће од 0;
- **"Field address is missing."** уколико поље **address** није присутно;
- **"Invalid adress."** уколико поље **address** не представља валидан рачун;

Одговарајуће провере се врше у наведеном редоследу.

### • Потврда доставе

Адреса /delivered

Тип POST

Заглављ Заглавље и њихов садржај су:

e

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

Вредност **<ACCESS\_TOKEN>** је стринг који представља JSON веб токен за приступ који је издат купцу приликом пријаве.

Тело Тело захтева је JSON објекат следећег формата:

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

Поље **id** представља идентификатор пристигле наруџбине.

Одговор Уколико су сва тражена заглавља и поља присутна и испуњавају наведена ограничења, резултат захтева је промена стања одговарајуће наруџбине, **трансфер средстава на рачуне курира и власника** и одговор са статусним кодом 200 без садржаја. **Трошкове трансфера сноси власник.**

У случају да заглавље недостаје, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

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

У случају да неко поље недостаје или је некоректно, резултат захтева је одговор са статусним кодом 400 и JSON објектом следећег формата:

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

Садржај поља `message` је:

- **"Missing order id."** уколико поље `id` недостаје;
- **"Invalid order id."** уколико поље `id` не представља цео број већи од 0 или у бази не постоји наруџбина са датим идентификатором или дату наруџбину није преузео неко од курира;
- **"Delivery not complete."** уколико ниједан од курира није купио дату наруџбину;

Одговарајуће провере се врше у наведеном редоследу.

Приликом реализације наплата потребно је дорадити следеће функционалности контејнера који намењен за рад са куририма (промене су обојене сивом бојом):

- **Преузимање наруџбине**

Адреса /pick\_up\_order

Тип POST

Заглавље Заглавље и њихов садржај су:  
е

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

Вредност `<ACCESS_TOKEN>` је стринг који представља JSON веб токен за приступ који је издат куриру приликом пријаве.

Тело Тело захтева је JSON објекат следећег формата:

```
{
  "id": 1,
  "address": "..."
}
```

Поље `id` представља идентификатор наруџбине коју курир жели да преузме и достави купцу. **Поље `address` представља рачун курира.**

Одговор Уколико су сва тражена заглавља присутна и сви тражени подаци присутни у телу захтева и испуњавају наведена ограничења, резултат захтева је промена стања одговарајуће наруџбине, **везивање курира за одговарајући паметни уговор** и одговор са статусним кодом 200 без икаквог садржаја.

У случају да заглавље недостаје, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

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

У случају да неко поље недостаје или је некоректно, резултат захтева је одговор са статусним кодом 400 и JSON објектом следећег формата:

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

Садржај поља **message** је:

- **"Missing order id."** уколико поље **id** недостаје;
- **"Invalid order id."** уколико поље **id** не представља цео број већи од 0, уколико бази не постоји наруџбина са датим идентификатором или је већ достављена;
- **"Missing address."** уколико поље **address** није присутно;
- **"Invalid address."** уколико поље **address** не представља валидан рачун;
- **"Transfer not complete."** уколико купац није извршио трансфер новца на уговор који одговара наведеној наруџбини;

Одговарајуће провере се врше у наведеном редоследу.

Приликом реализације наплата потребно је додати следеће функционалности контејнеру који намењен за рад са купцима:

- **Плаћање**

Адреса      /generate\_invoice

Тип          POST

Заглавље    Заглавље и њихов садржај су:  
е

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

Вредност **<ACCESS\_TOKEN>** је стринг који представља JSON веб токен за приступ који је издат купцу приликом пријаве.

Тело          Тело захтева је JSON објекат следећег формата:

```
{  
  "id": 1,  
  "address": "..."  
}
```

Поље **id** представља идентификатор наруџбине за коју се врши наплата.  
Поље **address** представља рачун купца.

## Одговор

Уколико су сва тражена заглавља и поља присутна и испуњавају наведена ограничења, резултат захтева је одговор са статусним кодом 200 са следећим JSON објектом:

```
{  
    "invoice": "....."  
}
```

Поље **invoice** представља трансакцију коју купац треба да изврши да би платио наруџбину.

У случају да заглавље недостаје, резултат је одговор са статусним кодом 401 и JSON објектом следећег формата и садржаја:

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

У случају да неко поље недостаје или је некоректно, резултат захтева је одговор са статусним кодом 400 и JSON објектом следећег формата:

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

Садржај поља **message** је:

- **"Missing order id."** уколико поље **id** недостаје;
- **"Invalid order id."** уколико поље **id** не представља цео број већи од 0 или у бази не постоји наруџбина са датим идентификатором;
- **"Missing address."** уколико поље **address** није присутно;
- **"Invalid address."** уколико поље адресс не представља валидан рачун;
- **"Transfer already complete."** уколико је трансфер средстава на одговарајући уговор већ извршен;

Одговарајуће провере се врше у наведеном редоследу.