FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE D COIMBRA

Gonçalo Almeida, aluno Nº 2020218868
Guilherme Branco, aluno Nº 2020216924
João Santos, aluno Nº 2020218995

Plataforma de Vendas Online

Projeto elaborado no âmbito da unidade
curricular de Bases de Dados

Maio de 2022

Índice

1. Manual do Utilizador

a. Descrição do sistema

Esta plataforma permite a comercialização de diferentes tipos de equipamentos eletrónicos (“computers”, “televisions” e “smartphones”). Cada produto é vendido por uma empresa específica e é caracterizado por um identificador único (id) e outros atributos específicos do produto além dos genéricos “name”, “stock”, “description” e “price”.

Sempre que há atualização dos detalhes de um produto é criada uma nova versão que mantém os detalhes não alterados e os novos, sendo assim criado um histórico das versões anteriores.

b. Níveis de acesso dos utilizadores

O tipo de utilizador com maior poder hierárquico é “admin”, com permissões de moderação e gestão da plataforma. Apenas um “admin” pode criar vendedores (“sellers”) e outros “admins” e criar campanhas promocionais. Outros tipos de utilizadores também têm acesso a operações exclusivas:

Um user do tipo “seller” pode adicionar e atualizar os produtos que vende.

Um user do tipo “buyer” pode efetuar compras, deixar feedback (“rating”) de um produto que tenha comprado e subscrever a campanhas promocionais.

Qualquer utilizador pode consultar estatísticas da plataforma relativas a vendas e campanhas promocionais.

c. *Endpoints* disponíveis

- Para todos os utilizadores

i. Registo de utilizadores

Qualquer pessoa pode registar-se como “buyer” sem ter realizado *login*. Apenas os administradores podem criar “sellers” e outros “admins”.

POST http://localhost:8080/dbproj/user

Payload JSON para “buyer”:

```
{
    "username": "username",
    "email": "mail@mail.com",
    "password": "password",
    "type": "buyers",
    "nif": 123456789,
    "home_addr": "Addr St. 20"
}
```

Payload JSON para “seller”:

```
{
    "username": "username",
    "email": "mail@mail.com",
    "password": "password",
    "type": "sellers",
    "nif": 123456789,
    "shipping_addr": "Addr St. 30"
}
```

Payload JSON para “admin”:

```
{
    "username": "username",
    "email": "mail@mail.com",
    "password": "password",
    "type": "admins"
}
```

ii. Autenticação

Fornecendo os dados de *login* irá receber um *token* de autenticação válido durante 20 minutos que deve ser enviado em pedidos seguintes no *header* “Authorization” com o valor “Bearer {colocar *token* aqui}”.

PUT http://localhost:8080/dbproj/user

Payload JSON:

```
{
    "username": "username",
    "password": "password"
}
```

Resultado:

```
{
    "status": 200,
    "token": {token}
}
```

iii. Consultar informações de um produto

Permite obter o nome, o stock, a descrição, o histórico de preços por versão, a média do rating e os comentários de *feedback* feitos ao produto com ID {product_id}.

GET http://localhost:8080/dbproj/product/{product_id}

Resultado:

```
{
    "results": {
        "comments": [
            "Comment 1",
            "Comment 2",
              [...]
        ],
        "description": "generic product description",
        "name": "This Product's Name",
        "prices": [
            "{price value} - YYYY-MM-DD HH:MM:SS",
            "{price value} - YYYY-MM-DD HH:MM:SS",
              [...]
        ],
        "rating": "{média de classificação}",
        "stock": {número de artigos disponíveis}
    },
```

iv. Deixar comentário/pergunta

Permite deixar um comentário/pergunta em relação a um produto com ID {product_id}.

POST http://localhost:8080/dbproj/questions/{product_id}

É possível também responder a uma pergunta com ID {parent_question_id} no produto, usando a seguinte alternativa:

POST http://localhost:8080/dbproj/questions/{product_id}/{parent_question_id}

Payload JSON:	Resultado:

`{`
`  "question": "{texto do comentário}"`
`}`
	`{`
	`  "status": 200,`
	`  "results": {ID do comentário}`
	`}`

v. Obter estatísticas mensais de vendas dos últimos 12 meses

Permite obter os detalhes das vendas mensais dos últimos 12 meses: o número de encomendas e o seu custo total para cada mês.

GET http://localhost:8080/proj/report/year

Resultado:

```
{
    "results": [
        {
            "month": "MM-YYYY",
            "orders": {n° de compras},
            "total_value": "{valor total}"
        },
        {
            "month": "MM-YYYY",
            "orders": {n° de compras},
            "total_value": "{valor total}"
        },
        [...]
    ],
    "status": 200
}
```

vi. Obter estatísticas de descontos aplicados por campanha

Permite obter uma lista das campanhas com a sua informação: o número de cupões emitidos, utilizados, e o valor total dos descontos aplicados.

GET http://localhost:8080/dbproj/report/campaign

Resultado:

```
{
    "results": [
        {
            "campaign_id": {ID da campanha},
            "generated_coupons": {nº de cupões gerados},
            "total_discount_value": {valor total de
descontos aplicados},
            "used_coupons": {nº de cupões usados}
        },
        [...]
    ],
    "status": 200
}
```

vii. Consultar notificações

Permite ao utilizador consultar as notificações que lhe são dirigidas.

GET http://localhost:8080/dbproj/inbox

Resultado:

```
{
    "results": [
        [
            "WEEKDAY, DD MONTH YYYY HH:MM:SS GMT",
            {ID da notificação},
            "{Texto da notificação}"
        ],
        [
            "WEEKDAY, DD MONTH YYYY HH:MM:SS GMT",
            {ID da notificação},
            "{Texto da notificação}"
        ],
       [...]
    ],
    "status": 200
}
```

- ## Para administradores

i. Criar campanha promocional

Permite criar uma campanha, tendo em conta que nunca pode existir mais que uma campanha ativa. Os compradores devem subscrever à campanha dentro do intervalo no qual está ativa para obterem cupões válidos durante 30 dias.

POST http://localhost:8080/dbproj/campaign

Payload JSON:

```
{
    "description": "{descrição da campanha}",
    "date_start": "YYYY-MM-DD",
    "date_end": "YYYY-MM-DD",
    "coupons": {nº de cupões},
    "discount": {% de desconto}
}
```

- ## Para vendedores

i. Adicionar um produto

Permite colocar um produto para venda, sendo que é necessário fornecer todos os seus detalhes.

POST http://localhost:8080/dbproj/product

Payload JSON:

```
{
    "description": "{descrição}",
    "type": "{tipo de produto}",
    "price": {preço unitário},
    "stock": 6{nº de itens em stock},
    "name": "{nome do produto}",

    {"attribute": "value" para outros
detalhes}
}
```

Resultado:

```
{
    "results": "{ID do produto}",
    "status": 200
}
```

ii. Atualizar informação de um produto

Permite atualizar os detalhes de um produto com ID {product_id}. Os novos detalhes são adicionados à base de dados com um novo atributo de versão.

PUT http://localhost:8080/dbproj/product/{product_id}

Payload JSON:

```
{
    {pares "detail_name": "{novo valor}"}
}
```

- ## Para compradores

i. Efetuar uma compra

Permite realizar uma compra. Se for incluído o ID de um cupão, o seu desconto é aplicado.

POST http://localhost:8080/dbproj/order

Payload JSON:

```
{
    "cart": [
        {
            "product_id": {ID do produto},
            "quantity": {nº de unidades}
        },
        {
            "product_id": {ID do produto},
            "quantity": {nº de unidades}
        },
       [...]
    ],
    "coupon": {ID do cupão} (opcional)
}
```

ii. Deixar rating/feedback

Permite atribuir um *rating* de 1 a 5 a um produto com ID {product_id} que tenha sido comprado.

POST **http://localhost:8080/dbproj/rating/{product_id}**

Payload:

```
{
    "rating": {valor de 1 a 5},
    "comment": "{comentário do rating"
}
```

iii. Subscrever campanha

Permite obter acesso a um cupão de desconto da campanha com ID {campaign_id}. Os cupões obtidos têm um prazo de 30 dias.

PUT **http://localhost:8080/dbproj/subscribe/{campaign_id}**

Resultado:

```
{
    "results": {
        "coupon_id": {ID do cupão},
        "expiration_date": "YYYY-MM-DD HH:MM:SS"
    },
    "status": 200
}
```

2. Manual de Instalação

a. Pré-requisitos

Nesta secção é apresentada uma lista de pré-requisitos de instalação que devem ser atendidos antes da instalação.

- ### Software necessário

i. PostgresSQL versão 14
ii. *Python* versão 3.9+
iii. Postman versão 9.18.2

- ### Python packages

Para a utilização do *script* "api.py" são necessários os seguintes packages:

i. Flask versão 2.1.2;
ii. psycopg2-binary versão 2.9.3;
iii. PyJWT versão 2.3.0;
iv. cryptography versão 37.0.2;

b. Processo de instalação

Após verificar que reúne todos os requisitos, pode proceder à instalação. Para criar a DB deve executar o *script* "dbproj_create.sql" na linha de comandos da seguinte forma:

```
psql -h localhost -U postgres -f dbproj_create.sql postgres
```

Este *script*:

- Cria a DB com o nome "dbproj";
- Cria um utilizador com o nome "projuser" e a password "projuser";
- Cria as tabelas e funções;
- Insere alguns dados-exemplo nas tabelas.

c. Desinstalação

Para desinstalar a DB "dbproj" e remover o utilizador "projuser", execute o *script* "dbproj_drop.sql":

```
psql -h localhost -U postgres -f dbproj_drop.sql postgres
```

3. Pormenores da implementação

a. Segurança da base de dados

A palavra-passe usada para aceder à base de dados encontra-se encriptada usando uma chave fernet fornecida num ficheiro. É desencriptada em cada momento de conexão.

O utilizador criado para aceder à base de dados tem permissões limitadas: apenas pode conectar-se e realizar as operações *select, update* e *insert*, que são as necessárias para as funcionalidades implementadas. Foi também certificado que novos utilizadores criados não adquirem automaticamente permissões.

As *passwords* dos utilizadores são armazenadas na base de dados de forma encriptada.

b. Autenticação de utilizadores

A autenticação é realizada através de *tokens* JWT. Ao realizar o *login* é gerada uma *token* com o ID de utilizador, usando uma chave associada à aplicação Flask, com o momento de criação definido e com uma audiência definida como o nome de sessão da aplicação (o que torna a *token* descodificável apenas por essa sessão). É também definido um momento de expiração para a *token*, pelo que uma sessão de *login* de um utilizador é válida por 20 minutos.

A partir do *token*, ao ser incluído no *header* das chamadas, é extraído o ID de utilizador, usado para verificar se tem autorização para realizar a operação pedida.

c. Controlo de concorrência e transações

Tendo em conta a possibilidade de vários utilizadores acederem em simultâneo às mesmas informações da base de dados, foram implementados *locks* para evitar situações problemáticas.

Para casos em que é necessária informação de toda a tabela foi realizado "*lock table*" (por exemplo, ao obter o valor de "max(campaign_id)", é necessário evitar a inserção de novas linhas até ao fim da transação).

Ao realizar uma compra é feito *lock* da tabela "products" para evitar possíveis *deadlocks* quando compradores acedem aos mesmos produtos por ordens diferentes. Foi também tido em conta que a instrução "UPDATE" faz *lock* às linhas atualizadas implicitamente (por exemplo, ao subscrever uma campanha, o número de cupões é decrementado em 1, mas não há risco desta operação ser realizada na falta de cupões suficientes para vários compradores que tentam subscrever em simultâneo).

Finalmente, transações que apenas envolvem "SELECT"s foram definidas como "*read only*".

d. Notificações

Foram implementados *triggers* para notificar utilizadores em relação aos seguintes eventos:

- É colocada/respondida uma questão em relação a um produto;
  - Vendedor do produto e utilizador que escreveu a questão a ser respondida são notificados do ID da nova questão e do seu conteúdo.
- É realizada uma venda;
  - Vendedores cujos produtos foram comprados são notificados do ID da venda e do seu total.
- É deixado *feedback* (rating e comentário) em relação a um produto;
  - Vendedor do produto é notificado da classificação atribuída e comentário associado.

4. Plano de desenvolvimento

Atividades	**Gonçalo**	**Guilherme**	**João**
Registo de utilizadores			
Autenticação de utilizadores			
Criar novo produto			
Atualizar detalhes do produto			
Efetuar compra			
Deixar rating/feedback			
Deixar comentário/pergunta			
Consultar informações genéricas de um produto			
Obter estatísticas mensais de vendas			
Criar nova campanha			
Subscrever campanha			
Obter estatísticas de campanha			
Notificações			
Elaboração do relatório final			
Total de horas:			