Project RESTful API :: Ecommerce

Setup dan Konfigurasi

Database

- Buatlah sebuah database baru
- Restore file ecommerce-db.sql ke dalam database yang baru saja dibuat
- Copy file .env.example ke .env, lalu ubah `DATABASE_NAME`, `DATABASE_HOST`, `DATABASE_USERNAME` dan `DATABASE_PASSWORD` sesuai kebutuhan

Logging

- Ubah `LOG_LEVEL` untuk mendapatkan level log sesuai yang diinginkan

Debug	Semua query ke database (SQL)
Info	Pengaksesan route, handler, service, repository
Error	Semua error dari eksekusi database query, return dari service,

- Output log ada di file app.log di dalam folder /log. Ubah `LOG_PATH` untuk menyimpan di folder yang lain. `LOG_PATH` harus diawali dengan karakter "/".

Session

- User yang berhasil login, akan mendapat sebuah token yang bisa digunakan untuk mengakses protected route. Secara bawaan (by default), token ini tidak memiliki batas waktu kedaluwarsa.
- Ubah `SESSION_LIFETIME` jika ingin menetapkan batas waktu kedaluwarsa. `SESSION_LIFETIME` menggunakan satuan menit. Ganti dengan 0 (angka nol) jika ingin mematikan fitur ini.

HTTP Server

- HTTP Server, secara bawaan, akan melayani request pada port 8080. Ubah `PORT` jika ingin mengganti port bawaan tersebut.

Instruksi Penggunaan

- Gunakan tool Postman untuk melakukan review
- Postman collection tersedia di dalam source code repository. Gunakan menu File | Import dan memilih file 20241118Project.postman_collection.json

Catatan

- Semua kalkulasi dilakukan on-the-fly. (Best Seller, New Product, Product Rating, Discount Rate / Net Price, dll) Tidak ada yang menggunakan tabel agregat atau CRON atau prekalkulasi.
- Validasi input hanya menggunakan package go-playground/validator/v10 (baked-in dan custom validation). Pesan error diterjemahkan ke Bahasa Indonesia menggunakan package go-playground/universal-translator + go-playground/locales. Tidak ada validasi manual.

API Endpoints

a. Auth

1. POST /api/register

Validasi

Name	required, max=30
Username	required, unique, max=40
Password	required, max=40

Contoh request

```
{
    "name": "Customer Satu",
    "username": "customer1@mail.com",
    "password": "drowssap"
}
```

Contoh response sukses

```
{
    "status_code": 201,
    "message": "Customer registered successfully.",
    "data": {
        "name": "Customer Satu",
        "username": "customer1@mail.com",
        "created_at": "2024-11-24T23:16:09.850622+07:00"
    }
}
```

Contoh response gagal

```
{
    "status_code": 422,
    "message": "invalid input",
    "data": [
        {
            "field": "name",
            "message": "name wajib diisi",
            "tag": "required",
            "value": "",
            "param": ""
        }
    ]
}
```

2. POST /api/login

Validasi

Username	required
Password	required

Contoh request

```
{
    "username": "customer1@mail.com",
    "password": "drowssap"
}
```

Contoh response sukses

```
{
    "status_code": 200,
    "message": "user authenticated",
    "data": {
        "token": "f29523d7-1df6-42df-b39d-42b34a958944",
        "expired_at": "6107-12-19T01:41:41.210877+07:00"
    }
}
```

Contoh response gagal

```
{
    "status_code": 401,
    "message": "invalid username and/or password."
}
```

b. Home

1. GET /api/banners

Contoh response sukses

```
{
    "status_code": 200,
    "message": "Get all banners",
    "data": [
        {
            "photo": "https://placehold.co/600x400?text=Banner+1",
            "title": "Banner Title 1",
            "subtitle": "Banner Subtitle 1",
            "path_page": "https://excitable-circumference.info"
        },
        {
            "photo": "https://placehold.co/600x400?text=Banner+5",
            "title": "Banner Title 5",
            "subtitle": "Banner Subtitle 5",
            "path_page": "https://flowery-abacus.biz/"
        }
    ]
}
```

2. GET /api/best-sellers

Contoh response sukses

```
{
    "status_code": 200,
    "page": 1,
    "limit": 6,
    "total_items": 42,
    "total_pages": 7,
    "data": [
        {
            "id": 2,
            "name": "Product 2",
            "thumbnail_url": "",
            "price": "",
            "discount_rate": null,
            "net_price": "",
            "category_id": 8,
            "rating": null,
            "on_my_wishlist": null,
            "created_at": "",
            "is_new": false
        },
        {
            "id": 12,
            "name": "Product 12",
            "thumbnail_url": "",
            "price": "",
            "discount_rate": null,
            "net_price": "",
            "category_id": 9,
            "rating": null,
            "on_my_wishlist": null,
            "created_at": "",
            "is_new": false
        }
    ]
}
```

3. GET /api/categories

Contoh response sukses

```
{
    "status_code": 200,
    "message": "Get all categories",
    "data": [
        {
            "id": 1,
            "name": "Category A"
        },
        {
            "id": 20,
            "name": "Category T"
        }
    ]
}
```

4. GET /api/weeklies

Contoh response sukses

```
{
    "status_code": 200,
    "message": "user authenticated",
    "data": {
        "token": "f29523d7-1df6-42df-b39d-42b34a958944",
        "expired_at": "6107-12-19T01:41:41.210877+07:00"
    }
}
```

5. GET /api/products

Contoh response sukses

```
{
    "status_code": 200,
    "message": "user authenticated",
    "data": {
        "token": "f29523d7-1df6-42df-b39d-42b34a958944",
        "expired_at": "6107-12-19T01:41:41.210877+07:00"
    }
}
```

6. **GET /api/recommendations**

7. **POST /api/wishlist**

8. **DELETE /api/wishlist**

9. **GET /api/cart**

c. Product

1. **GET /api/products**

2. **POST /api/cart**

3. **GET /api/categories/{id}/products**

d. Checkout

1. **GET /api/product/{id}**

2. **GET /api/cart**

3. **PUT /api/**

4. **DELETE /api/**

5. **GET /api/address**

6. **POST /api/order**

e. Account

1. **GET /api/profile**

2. **PUT /api/profile**

3. **POST /api/address**

4. **GET /api/address**

5. **PUT /api/address/{id}/default**

6. **PUT /api/address/{id}**

7. **DELETE /api/address**