

AppPayment

Halo 🙌 Selamat Datang di Repository Payment Gateway Saya. AppPayment adalah sebuah implementasi dan integrasi payment gateway yang dibangun menggunakan Laravel 9 dalam bentuk API.

Creator: Mohammad Arfan Maulana

API ini tidak hanya mencakup payment gateway checkout dan pembayaran saja, namun juga mencakup pengelolaan event (membuat, mengupdate, menghapus, melihat daftar event), dan juga kelola transaksi (membatalkan, melihat daftar transaksi)

Tools yang digunakan dalam pengembangan

| Tools | Version |
|-------------------------------------|---------|
| Composer | 2.1.14 |
| Midtrans core API (Payment gateway) | - |
| Laravel | 9.11 |
| PHP | 8.0.2 |
| Laravel Sanctum (Authentication) | 2.15 |
| Laragon | 5.0 |
| Postman | - |

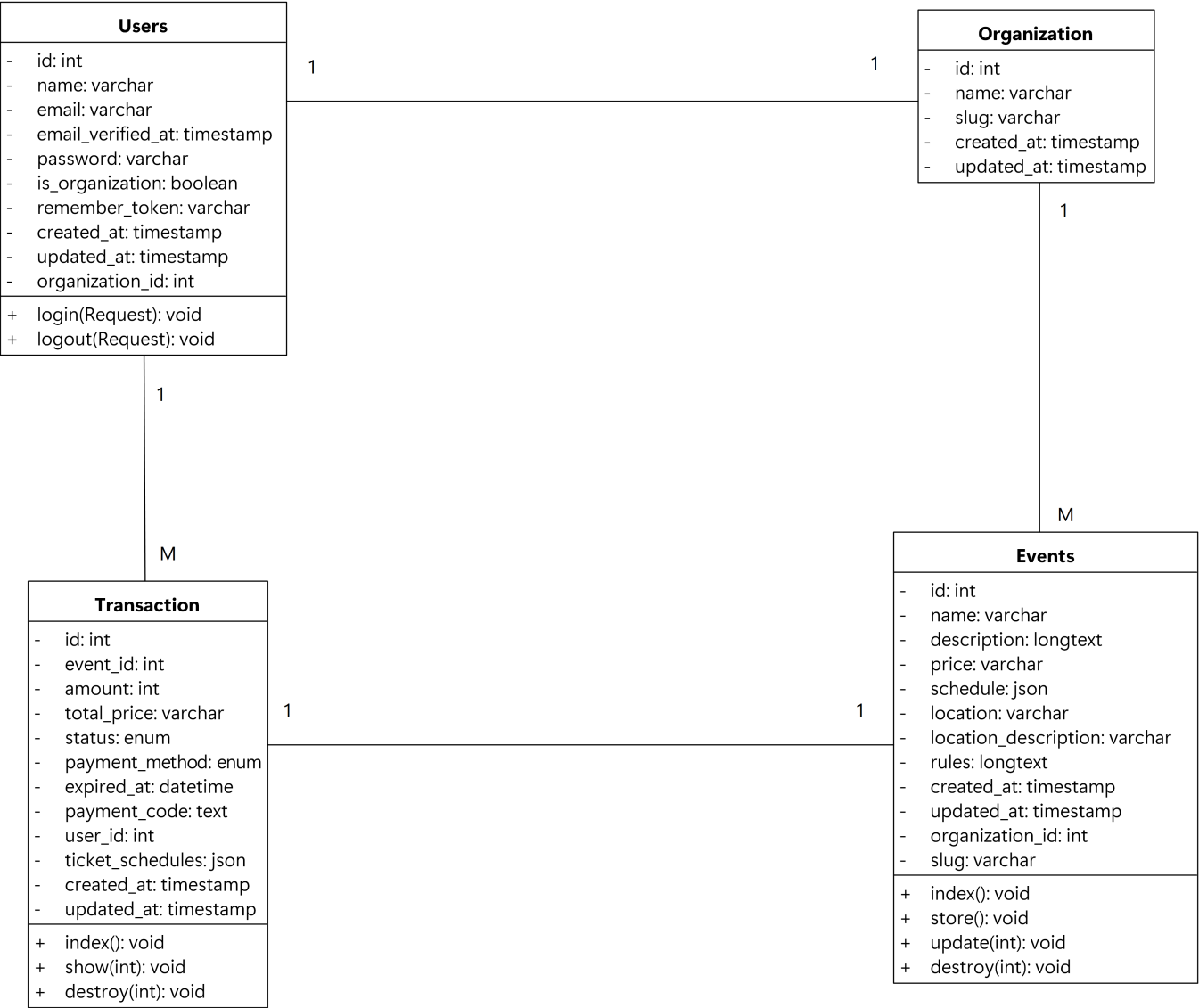
Instalasi

- Klon Repository ini
- Ekstrak file yang telah didownload
- Jalankan terminal, dan arahkan terminal ke direktori ekstrak file menggunakan `cd pathtodirectory`
- Dalam terminal jalankan `composer install`
- Ubah nama file `.env.example` file ke `.env`
- Edit file `.env` pada `DB_CONNECTION`, `DB_HOST`, `DB_PORT`, `DB_USERNAME`, `DB_PASSWORD` sesuaikan dengan konfigurasi database anda
- Didalam terminal jalankan `php artisan migrate`
- Lalu jalankan `php artisan db:seed` untuk mengisi database dengan data default
- Selanjutnya, dalam terminal jalankan `php artisan serve`

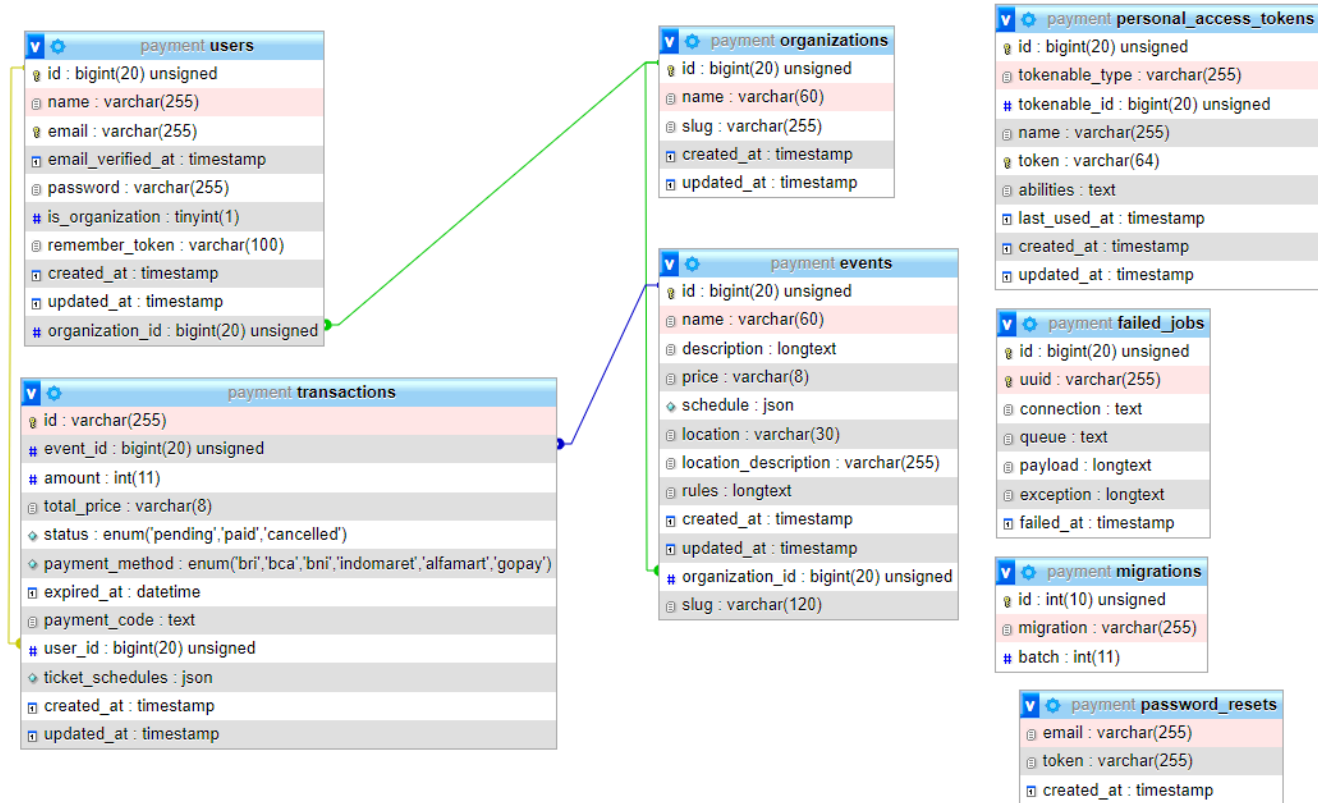
Selesai, API berjalan pada localhost:8000/api

Struktur Basis Data

Class Diagram



PhpMyAdmin Designer



Autentikasi

Users

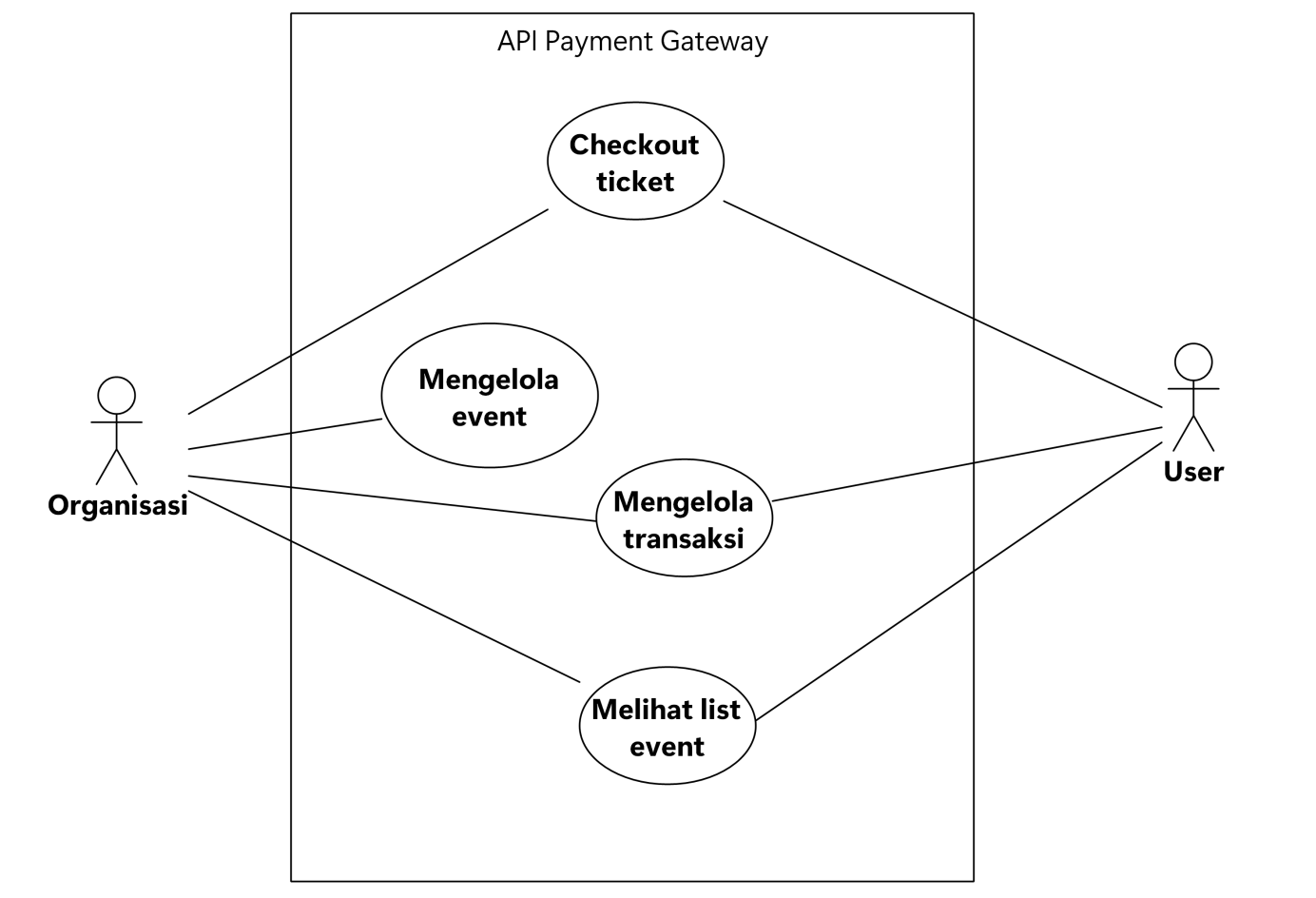
Ada dua tipe user pada API ini, diantaranya

1. User biasa
2. User organisasi

Pembeda dari dua user tersebut ada pada atribut **organization_id** dan **is_organization** pada tabel *users*.

Pada user organisasi, atribut *is_organization* mengandung value true atau nilai 1, dan memiliki relasi dengan entitas **Organization**, dihubungkan dengan *foreign key organization_id*

Berikut adalah use case diagram untuk menggambarkan apa saja yang dapat dilakukan oleh masing-masing user.



Kredensial

Berikut adalah email dan password default untuk melakukan login pada API

| Email | Password | Role |
|----------------------|----------|-----------------|
| user@test.com | test123 | User biasa |
| kemendikbud@test.com | test123 | User organisasi |

Login

Sebelum menggunakan fitur lain, diharuskan login terlebih dahulu

| Type | Endpoint |
|------|--------------------------|
| POST | localhost:8000/api/login |

Request Format (Body)

| Name | Description |
|----------|-------------|
| email | required |
| password | required |

Sample Response

Berikut adalah contoh response yang akan didapatkan jika login berhasil, silahkan simpan *access_token* pada header, untuk tipe authorization pilih *Bearer*. Token tersebut digunakan untuk mengakses fitur yang membutuhkan login.

```
{
  "status": "success",
  "message": "Login successfully",
  "errors": null,
  "content": {
    "status_code": 200,
    "access_token": "2|pyzsEI59uRqnKFe0NxVMJK9BZH2CnfnSNyy5uUNy",
    "token_type": "Bearer"
  }
}
```

Contoh

Berikut screenshot contoh request login menggunakan aplikasi postman

The screenshot displays the Postman interface for a POST request to `localhost:8000/api/login`. The request body is `x-www-form-urlencoded` with the following data:

| KEY | VALUE | DESCRIPTION |
|----------|----------------------|-------------|
| email | kemendikbud@test.com | |
| password | test123 | |

The response is shown in the 'Body' tab, displaying a JSON object with a status of 'success' and an access token. The status bar indicates a 200 OK response with a time of 6.20 s and a size of 484 B.

```
1 {
2   "status": "success",
3   "message": "Login successfully",
4   "errors": null,
5   "content": {
6     "status_code": 200,
```

Logout

Setelah selesai melakukan aktivitas, user dapat melakukan logout

| Type | Endpoint |
|------|---------------------------|
| POST | localhost:8000/api/logout |

Headers

| Key | Value |
|---------------|--------------------|
| Authorization | Bearer tokendisini |

Sample Response

Berikut adalah contoh response yang akan didapatkan jika logout berhasil, token yang digunakan sebelumnya tidak akan lagi berfungsi.

```
{
  "status": "success",
  "message": "Logout successfully",
  "errors": null,
  "content": null
}
```

Example

Berikut screenshot contoh request logout menggunakan aplikasi postman

localhost:8000/api/logout

POST localhost:8000/api/logout

Headers (9)

| KEY | VALUE | DESCRIPTION |
|----------------|---|-------------|
| Authorization | Bearer 2 pyzsEI59uRqnKFe0NxVMJK9BZH2Cn... | |
| Cookie | Cookie_1=value | |
| Postman-Token | <calculated when request is sent> | |
| Content-Length | 0 | |

Body

```
{
  "status": "success",
  "message": "Logout successfully",
  "errors": null,
  "content": null
}
```

Status: 200 OK Time: 923 ms Size: 388 B

Mendapatkan Daftar Event

User dapat melihat event apa saja yang terdapat pada sistem, berikut adalah cara mendapatkan daftar event, daftar event dilimit default 10 item menggunakan pagination

Type Endpoint

GET localhost:8000/api/events

Headers

Key Value

Authorization Bearer tokendisini

Sample Response

Berikut adalah contoh response yang akan didapatkan jika request berhasil

```
{
  "data": [
    {
      "id": 4,
      "name": "Pemanfaatan Data Mining pada Revolusi Industri 4.0",
      "description": "Memanfaatkan secara maksimal prosedur data mining",
      "price": "Rp. 50.000",
      "schedule": "[\"11-06-2022\", \"12-06-2022\"]",
      "location": "Tangerang Selatan",
      "location_description": "Depan Pasar Ciputat",
      "rules": "Mengenakan Masker dan faceshield",
      "organization": {
        "id": 1,
        "name": "Kemendikbud RI",
        "slug": "kemendikbud-ri",
        "created_at": "14-06-2022"
      }
    }
  ],
  "links": {
    "first": "http://localhost:8000/api/events?page=1",
    "last": "http://localhost:8000/api/events?page=1",
    "prev": null,
    "next": null
  },
  "meta": {
    "current_page": 1,
    "from": 1,
    "last_page": 1,
    "links": [
      {
        "url": null,
        "label": "&laquo; Previous",
        "active": false
      },
      {
        "url": "http://localhost:8000/api/events?page=1",
        "label": "1",

```

Example

localhost:8000/api/events

Save

GET

localhost:8000/api/events

Send

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Cookies

Headers

Hide auto-generated headers

| | KEY | VALUE | DESCRIPTION | ... | Bulk Edit | Presets |
|-------------------------------------|-----------------|---|-------------|-----|-----------|---------|
| <input checked="" type="checkbox"/> | Authorization ⓘ | Bearer 1 UE3Dza4pVPagvbmTeVbaoA6OTYAK1... | | | | |
| <input checked="" type="checkbox"/> | Cookie ⓘ | Cookie_1=value | | | | |
| <input type="checkbox"/> | Content-Type ⓘ | application/json | | | | |

Body

Cookies (1)

Headers (10)

Test Results

Status: 200 OK

Time: 741 ms

Size: 1.8 KB

Save Response

Pretty

Raw

Preview

Visualize

JSON

```

1  {
2    "data": [
3      {
4        "id": 3,
5        "name": "Iste quos pariaturs reprehenderit doloribus voluptas.",
6        "description": "Quo repellendus quisquam ut ut voluptatibus impedit. Vitae rem modi quos et sunt eaque. Et quos voluptatem est.",
7        "price": "Rp. 311.440",
8        "schedule": "\"[\\\"1995-05-23\\\"],[\\\"1985-12-20\\\"]\"",

```

Membuat Event

Jika user yang digunakan adalah tipe **user organisasi**, user dapat membuat event baru, berikut adalah cara dan format membuat event baru

| Type | Endpoint |
|------|----------|
|------|----------|

| Type | Endpoint |
|------|---------------------------|
| POST | localhost:8000/api/events |

Request Format (Body)

| Name | Description |
|----------------------|-------------------|
| name | required, min 3 |
| description | required, min 5 |
| price | required, numeric |
| schedule[] | required |
| location | required |
| location_description | required |
| rules | required |

Headers

| Key | Value |
|---------------|--------------------|
| Authorization | Bearer tokendisini |

Sample Response

Berikut adalah contoh response yang akan didapatkan jika event berhasil dibuat

```
{
  "data": {
    "id": 4,
    "name": "Contoh nama event",
    "description": "deskripsi panjang event",
    "price": "Rp. 40.000",
    "schedule": "[\"11-06-2022\", \"12-07-2022\"]",
    "location": "Tangerang Selatan",
    "location_description": "Depan Pasar Ciputat",
    "rules": "Mengenakan Masker",
    "organization": {
      "id": 1,
      "name": "Kemendikbud RI",
      "slug": "kemendikbud-ri",
      "created_at": "14-06-2022"
    }
  },
  "success": true,
  "total": 2,
  "code": 206
}
```

Example

Berikut screenshot contoh request membuat event menggunakan aplikasi postman

The first screenshot shows a POST request to `localhost:8000/api/events` with the following body data:

| KEY | VALUE | DESCRIPTION |
|--|---|-------------|
| <input checked="" type="checkbox"/> name | Pemanfaatan Data Mining pada Revolusi Indust... | |
| <input checked="" type="checkbox"/> description | Memfaatkan secara maksimal prosedur data... | |
| <input checked="" type="checkbox"/> price | 40000 | |
| <input checked="" type="checkbox"/> schedule[] | 11-06-2022 | |
| <input checked="" type="checkbox"/> schedule[] | 12-07-2022 | |
| <input checked="" type="checkbox"/> location | Tangerang Selatan | |
| <input checked="" type="checkbox"/> location_description | Depan Pasar Ciputat | |
| <input checked="" type="checkbox"/> rules | Menggunakan Masker | |
| Key | Value | Description |

The second screenshot shows the Headers tab for the same request:

| Key | Value | Description |
|---|---|-------------|
| <input checked="" type="checkbox"/> Authorization ⓘ | Bearer 1 UE3Dza4pVPagvbmTeVbaoA6OTYAk1... | |
| <input checked="" type="checkbox"/> Cookie ⓘ | Cookie_1=value | |
| <input checked="" type="checkbox"/> Postman-Token ⓘ | <calculated when request is sent> | |
| <input checked="" type="checkbox"/> Content-Type ⓘ | application/x-www-form-urlencoded | |
| <input checked="" type="checkbox"/> Content-Length ⓘ | <calculated when request is sent> | |
| <input checked="" type="checkbox"/> Host ⓘ | <calculated when request is sent> | |
| <input checked="" type="checkbox"/> User-Agent ⓘ | PostmanRuntime/7.29.0 | |
| <input checked="" type="checkbox"/> Accept ⓘ | */* | |
| <input checked="" type="checkbox"/> Accept-Encoding ⓘ | gzip, deflate, br | |
| <input checked="" type="checkbox"/> Connection ⓘ | keep-alive | |
| Key | Value | Description |

Mengupdate Event

Jika user yang digunakan adalah tipe **user organisasi**, user dapat mengedit event, berikut adalah cara dan format mengedit event. Event yang akan diedit haruslah memiliki atribut `organization_id` yang sama dengan

user.

| Type | Endpoint |
|------|-------------------------------------|
| PUT | localhost:8000/api/events/{eventid} |

Request Format (Body)

| Name | Description |
|----------------------|-------------------|
| name | required, min 3 |
| description | required, min 5 |
| price | required, numeric |
| schedule[] | required |
| location | required |
| location_description | required |
| rules | required |

Headers

| Key | Value |
|---------------|--------------------|
| Authorization | Bearer tokendisini |

Sample Response

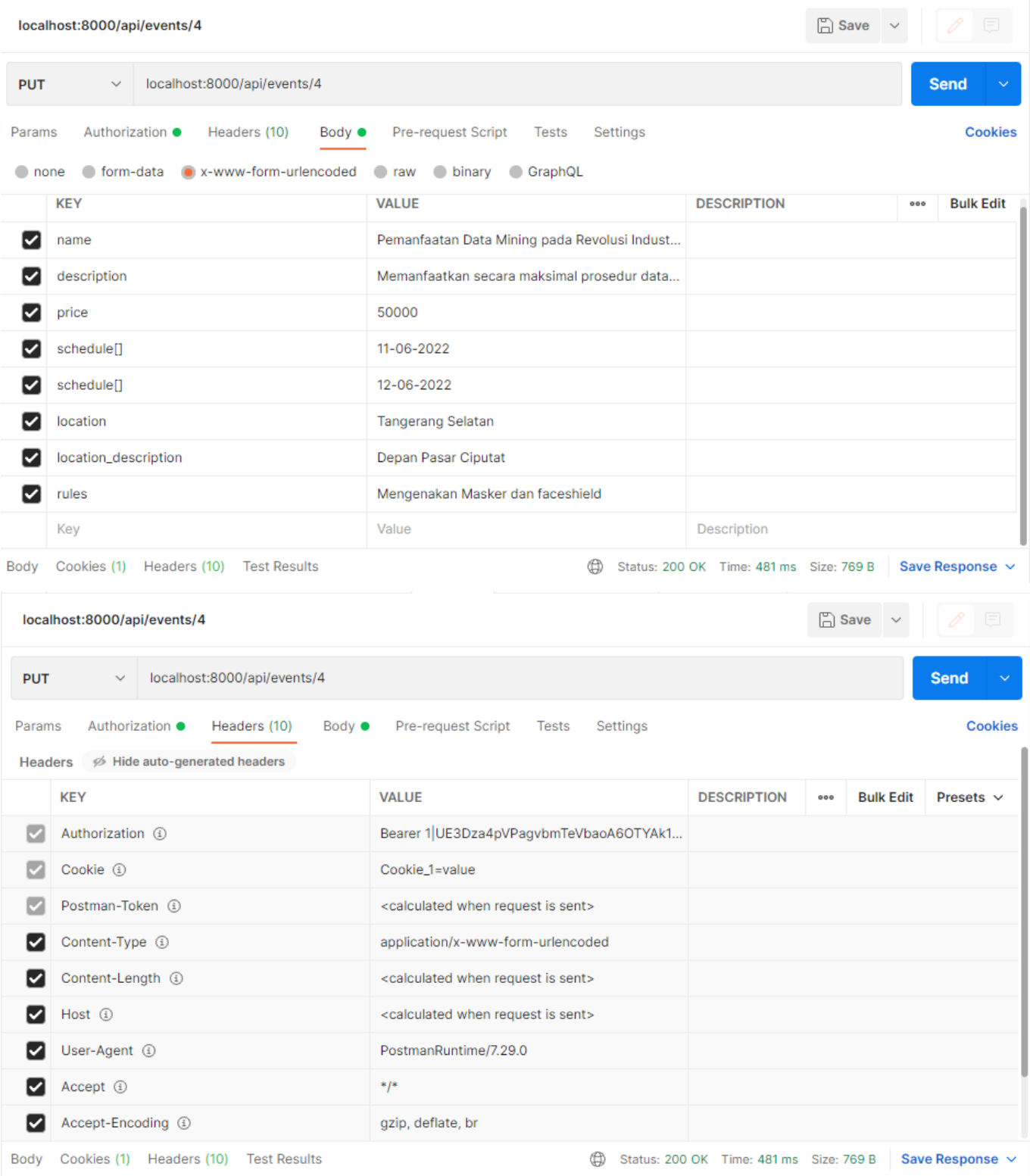
Berikut adalah contoh response yang akan didapatkan jika event berhasil diupdate.

```
{
  "data": {
    "id": 4,
    "name": "Contoh nama event",
    "description": "Memanfaatkan secara maksimal prosedur data mining",
    "price": "Rp. 50.000",
    "schedule": "[\"11-06-2022\", \"12-06-2022\"]",
    "location": "Tangerang Selatan",
    "location_description": "Depan Pasar Ciputat",
    "rules": "Mengenakan Masker dan faceshield",
    "organization": {
      "id": 1,
      "name": "Kemendikbud RI",
      "slug": "kemendikbud-ri",
      "created_at": "14-06-2022"
    }
  },
  "success": true,
  "total": 2,
```

```
"code": 206
}
```

Example

Berikut screenshot contoh request mengedit event menggunakan aplikasi postman



Menghapus Event

Jika user yang digunakan adalah tipe **user organisasi**, user dapat menghapus event, berikut adalah cara dan format mengedit event. Event yang akan dihapus haruslah memiliki atribut *organization_id* yang sama dengan user.

| Type | Endpoint |
|--------|-------------------------------------|
| DELETE | localhost:8000/api/events/{eventid} |

Headers

| Key | Value |
|---------------|--------------------|
| Authorization | Bearer tokendisini |

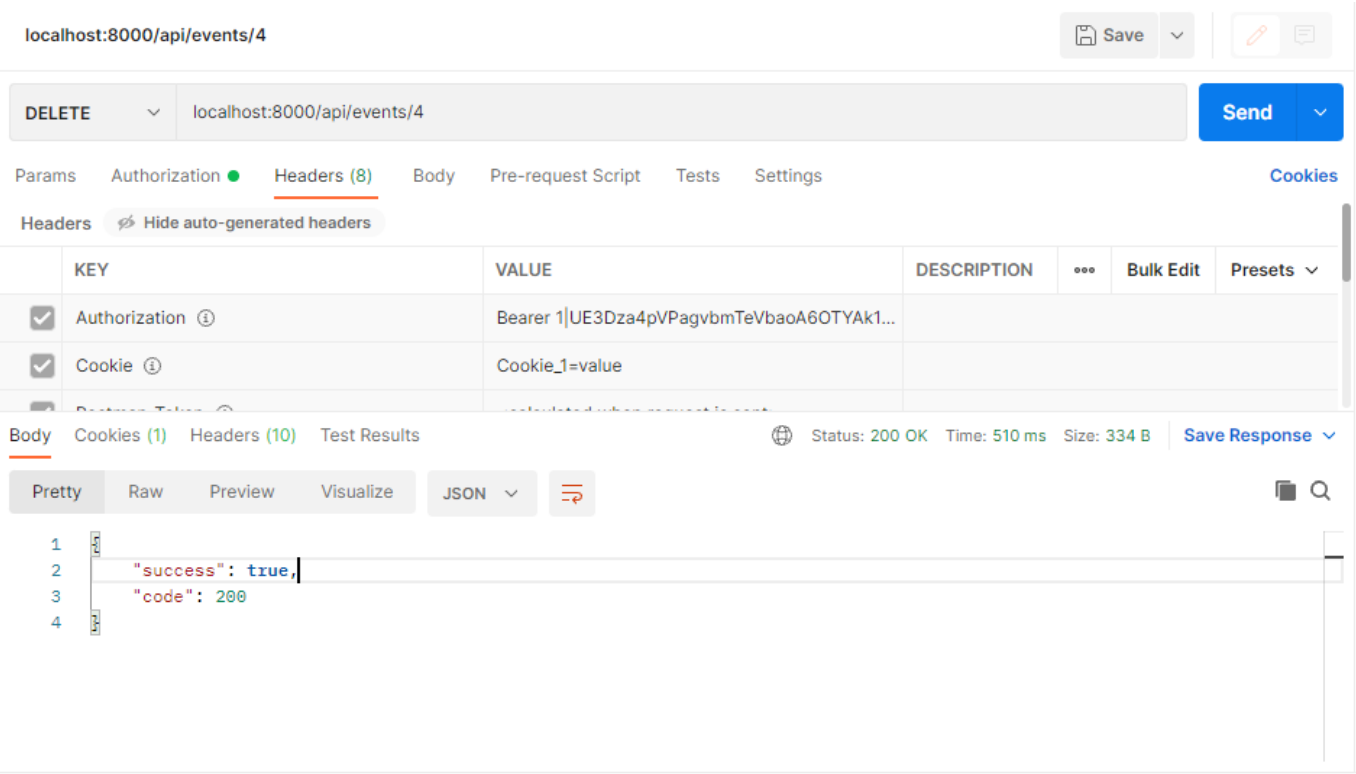
Sample Response

Berikut adalah contoh response yang akan didapatkan jika event berhasil dihapus.

```
{
  "success": true,
  "code": 200
}
```

Example

Berikut adalah contoh response menghapus event menggunakan aplikasi postman.



Checkout Event

Berikut adalah bagian checkout tiket event

| Type | Endpoint |
|------|-----------------------------|
| POST | localhost:8000/api/checkout |

Request Format (Body)

Pada event_id diharuskan sesuai dengan id event yang ada.

| Name | Description |
|--------------------|---|
| event_id | required, numeric, exists |
| amount | required, numeric |
| payment_method | required, enum:bni,bca,bri,indomaret,alfamart,gopay |
| ticket_schedules[] | required |

Headers

| Key | Value |
|---------------|--------------------|
| Authorization | Bearer tokendisini |

Sample Response

Berikut adalah contoh response yang akan didapatkan jika checkout tiket berhasil.

```
{
  "data": {
    "id": "order-1-1-1655548920",
    "event_id": "1",
    "amount": "2",
    "total_price": 80000,
    "status": "pending",
    "payment_method": "indomaret",
    "payment_code": "052179521795",
    "expired_at": "19-06-2022 10:42",
    "ticket_schedule": [
      "11-06-2022"
    ],
    "event": {
      "id": 1,
      "name": "Pemanfaatan Data Mining pada Revolusi Industri 4.0",
      "description": "Memanfaatkan secara maksimal prosedur data mining",
      "price": "Rp. 40.000",
      "schedule": "[\"11-06-2022\", \"12-07-2022\"]",
      "location": "Tangerang Selatan",
      "location_description": "Depan Pasar Ciputat",
      "rules": "Mengenakan Masker",
      "organization": {
```

```

      "id": 1,
      "name": "Kemendikbud RI",
      "slug": "kemendikbud-ri",
      "created_at": "18-06-2022"
    }
  },
  "success": true,
  "total": 1,
  "code": 206
}

```

Setelah berhasil checkout, transaksi akan terbuat namun pembayaran masih dalam status **pending**

Example

Berikut adalah contoh response checkout menggunakan aplikasi postman.

The screenshot shows a Postman interface for a POST request to `localhost:8000/api/checkout`. The request body is a JSON object with the following data:

| KEY | VALUE | DESCRIPTION |
|--|------------|-------------|
| <input checked="" type="checkbox"/> event_id | 1 | |
| <input checked="" type="checkbox"/> amount | 2 | |
| <input checked="" type="checkbox"/> payment_method | indomaret | |
| <input checked="" type="checkbox"/> ticket_schedules[] | 11-06-2022 | |
| Key | Value | Description |

The response is a 201 Created status, with a time of 3.60 s and a size of 989 B. The response body is a JSON object:

```

{
  "event_id": "1",
  "amount": "2"
}

```

Verifikasi atau Konfirmasi Pembayaran

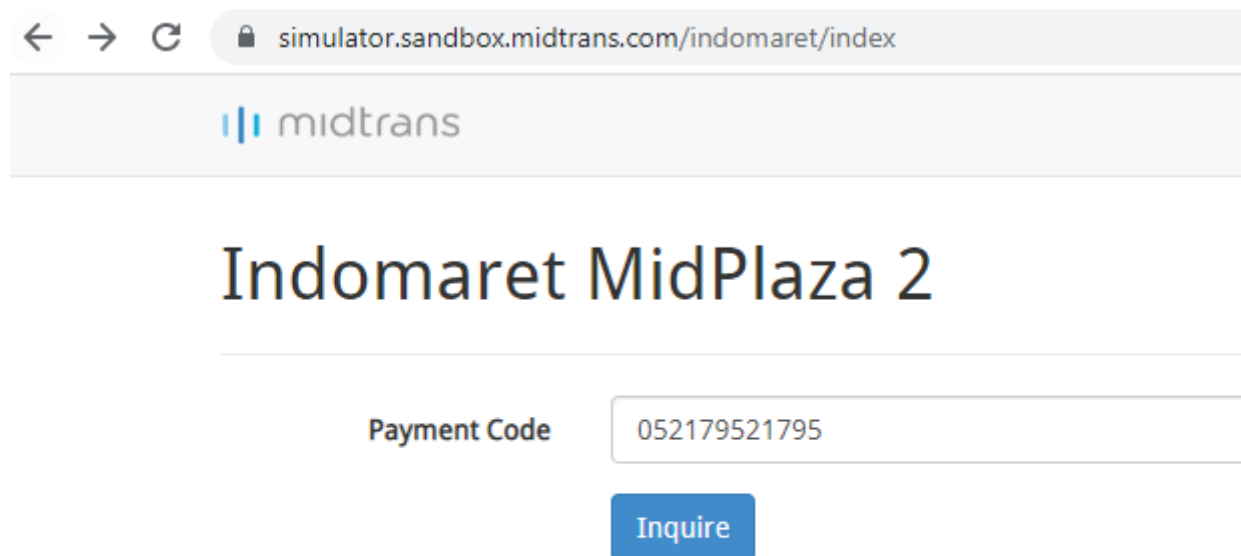
Sebelum melakukan konfirmasi atau verifikasi pembayaran ini, diharuskan terlebih dahulu untuk membayar atau menggunakan simulasi pembayaran yang disediakan oleh Midtrans sebagai berikut: (Sesuaikan dengan metode pembayarannya)

| Nama | Link |
|---------------------|---|
| Indomaret simulator | https://simulator.sandbox.midtrans.com/indomaret/index |
| Alfamart simulator | https://simulator.sandbox.midtrans.com/alfamart/index |

| Nama | Link |
|-------------------------------|---|
| BCA Virtual Account simulator | https://simulator.sandbox.midtrans.com/bca/va/index |
| BNI Virtual Account simulator | https://simulator.sandbox.midtrans.com/bni/va/index |
| BRI Virtual Account simulator | https://simulator.sandbox.midtrans.com/bri/va/index |

Cara penggunaan simulator


1. Ikuti sesuai petunjuk yang tertera. Pada halaman simulator, dalam contoh ini saya menggunakan metode pembayaran Indomaret, pada kolom payment code masukkan payment code yang didapat saat checkout.
2. Lalu tekan tombol **Inquire**



The screenshot shows a web browser window with the address bar displaying `simulator.sandbox.midtrans.com/indomaret/index`. Below the address bar is the Midtrans logo. The main heading of the page is "Indomaret MidPlaza 2". Underneath, there is a form with a label "Payment Code" and a text input field containing the value "052179521795". Below the input field is a blue button labeled "Inquire".

3. Setelah itu akan muncul detail tiket event yang dibeli, klik tombol **Pay**

[←](#) [→](#) [↻](#) [🔒 simulator.sandbox.midtrans.com/indomaret/inquiry](#)



Indomaret MidPlaza 2

| | |
|--------------------|--|
| Customer Name | kemendikbud ri null |
| Item Description | 2x pemanfaatan data mining pada revolusi industri 4.0 @40000 |
| Transaction Amount | 80000.00 |
| Expired At | 2022-06-19 17:42:02 |
| Message | payment created |
| Merchant ID | |
| Product ID | |

Pay

4. Akan muncul message success transaction. lalu lakukan konfirmasi pembayaran pada API

| Type | Endpoint |
|------|----------|
|------|----------|

| | |
|-----|--|
| GET | localhost:8000/api/payment/verify/{id} |
|-----|--|

Ket : **id** diisi dengan id yang didapat pada saat checkout, dalam format order-x-x-xxxxxx, contoh <localhost:8000/api/payment/verify/order-6-3-1655451787>

Headers

| Key | Value |
|---------------|--------------------|
| Authorization | Bearer tokendisini |

Sample Response

Berikut adalah contoh response yang akan didapatkan jika verifikasi pembayaran sukses.

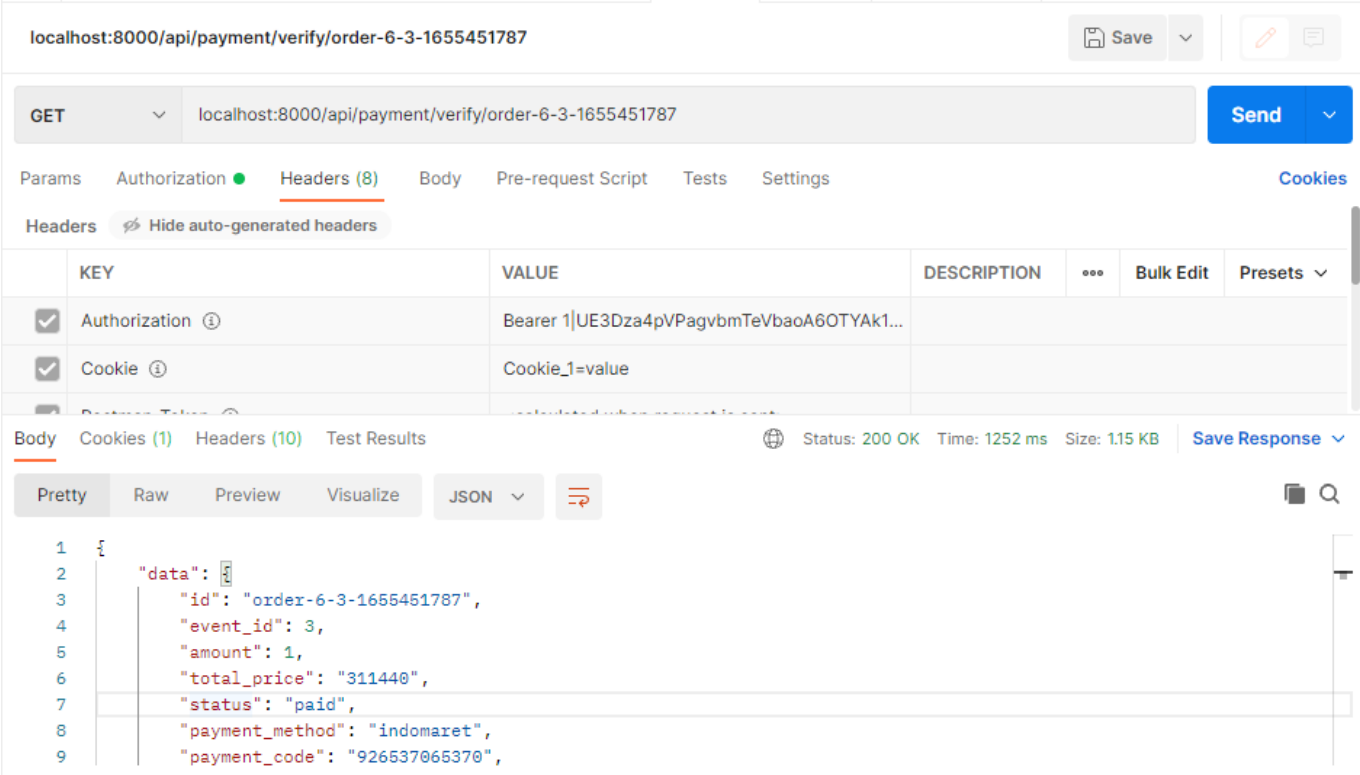
```
{
  "data": {
    "id": "order-1-1-1655548920",
    "event_id": 1,
    "amount": 2,
    "total_price": "80000",
```

```
    "status": "paid",
    "payment_method": "indomaret",
    "payment_code": "052179521795",
    "expired_at": "19-06-2022 10:42",
    "ticket_schedule": [
      "11-06-2022"
    ],
    "event": {
      "id": 1,
      "name": "Pemanfaatan Data Mining pada Revolusi Industri 4.0",
      "description": "Memanfaatkan secara maksimal prosedur data mining",
      "price": "Rp. 40.000",
      "schedule": "[\"11-06-2022\", \"12-07-2022\"]",
      "location": "Tangerang Selatan",
      "location_description": "Depan Pasar Ciputat",
      "rules": "Mengenakan Masker",
      "organization": {
        "id": 1,
        "name": "Kemendikbud RI",
        "slug": "kemendikbud-ri",
        "created_at": "18-06-2022"
      }
    }
  },
  "success": true,
  "total": 1,
  "code": 206
}
```

Dalam response tersebut, akan terlihat bahwa status akan berubah menjadi **paid** jika dalam kondisi sudah dibayar atau menggunakan simulator.

Example

Berikut adalah contoh response verifikasi pembayaran menggunakan aplikasi postman.



Melihat daftar transaksi

Fitur ini digunakan untuk melihat daftar transaksi user yang sedang login

| Type | Endpoint |
|------|--------------------------------|
| GET | localhost:8000/api/transaction |

Headers

| Key | Value |
|---------------|--------------------|
| Authorization | Bearer tokendisini |

Sample Response

Berikut adalah contoh response yang akan didapatkan jika daftar transaksi berhasil didapat.

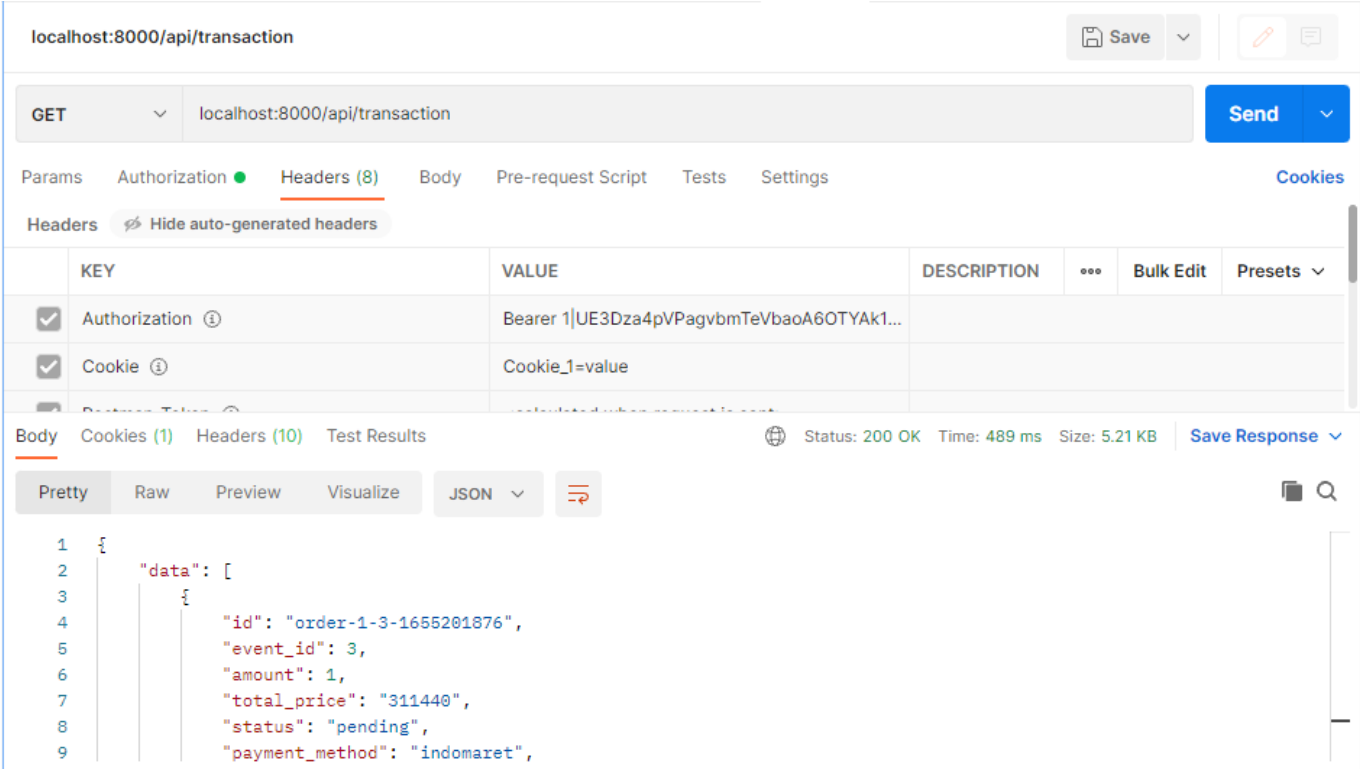
```
{
  "data": [
    {
      "id": "order-2-3-1655202009",
      "event_id": 3,
      "amount": 3,
      "total_price": "934320",
      "status": "pending",
      "payment_method": "gopay",
      "payment_code": null,
      "expired_at": "15-06-2022 10:20",
    }
  ]
}
```

```
"ticket_schedule": "\\[\\\\"1995-05-23\\",\\\\"1985-12-20\\")\\",
"event": {
  "id": 3,
  "name": "Iste quos pariatu reprehenderit doloribus voluptas.",
  "description": "Quo repellendus quisquam ut ut voluptatibus
impedit. Vitae rem modi quos et sunt eaque. Et quos voluptatem est.",
  "price": "Rp. 311.440",
  "schedule": "\\[\\\\"1995-05-23\\",\\\\"1985-12-20\\")\\",
  "location": "Reidview",
  "location_description": "Voluptatem doloremque.",
  "rules": "Consequatur voluptates iusto quo rerum molestiae.
Dolorum sit libero voluptatem ipsum. Corrupti voluptatem ut dolores fugiat dolorum
reiciendis quia.",
  "organization": {
    "id": 1,
    "name": "Kemendikbud RI",
    "slug": "kemendikbud-ri",
    "created_at": "14-06-2022"
  }
},
{
  "id": "order-6-3-1655451787",
  "event_id": 3,
  "amount": 1,
  "total_price": "311440",
  "status": "paid",
  "payment_method": "indomaret",
  "payment_code": "926537065370",
  "expired_at": "18-06-2022 07:43",
  "ticket_schedule": [
    "11-06-2022"
  ],
  "event": {
    "id": 3,
    "name": "Iste quos pariatu reprehenderit doloribus voluptas.",
    "description": "Quo repellendus quisquam ut ut voluptatibus
impedit. Vitae rem modi quos et sunt eaque. Et quos voluptatem est.",
    "price": "Rp. 311.440",
    "schedule": "\\[\\\\"1995-05-23\\",\\\\"1985-12-20\\")\\",
    "location": "Reidview",
    "location_description": "Voluptatem doloremque.",
    "rules": "Consequatur voluptates iusto quo rerum molestiae.
Dolorum sit libero voluptatem ipsum. Corrupti voluptatem ut dolores fugiat dolorum
reiciendis quia.",
    "organization": {
      "id": 1,
      "name": "Kemendikbud RI",
      "slug": "kemendikbud-ri",
      "created_at": "14-06-2022"
    }
  }
}
```

```
    ]
  }
```

Example

Berikut adalah contoh response daftar transaksi menggunakan aplikasi postman.



Melihat detail transaksi

User juga dapat melihat detail transaksi dengan id tertentu

| Type | Endpoint |
|------|-------------------------------------|
| GET | localhost:8000/api/transaction/{id} |

Keterangan : *id* diisi dengan id transaksi, dalam format order-x-x-xxxx, contoh `localhost:8000/api/transaction/order-6-3-1655451787`

Headers

| Key | Value |
|---------------|--------------------|
| Authorization | Bearer tokendisini |

Sample Response

Berikut adalah contoh response yang akan didapatkan jika detail transaksi berhasil didapatkan.

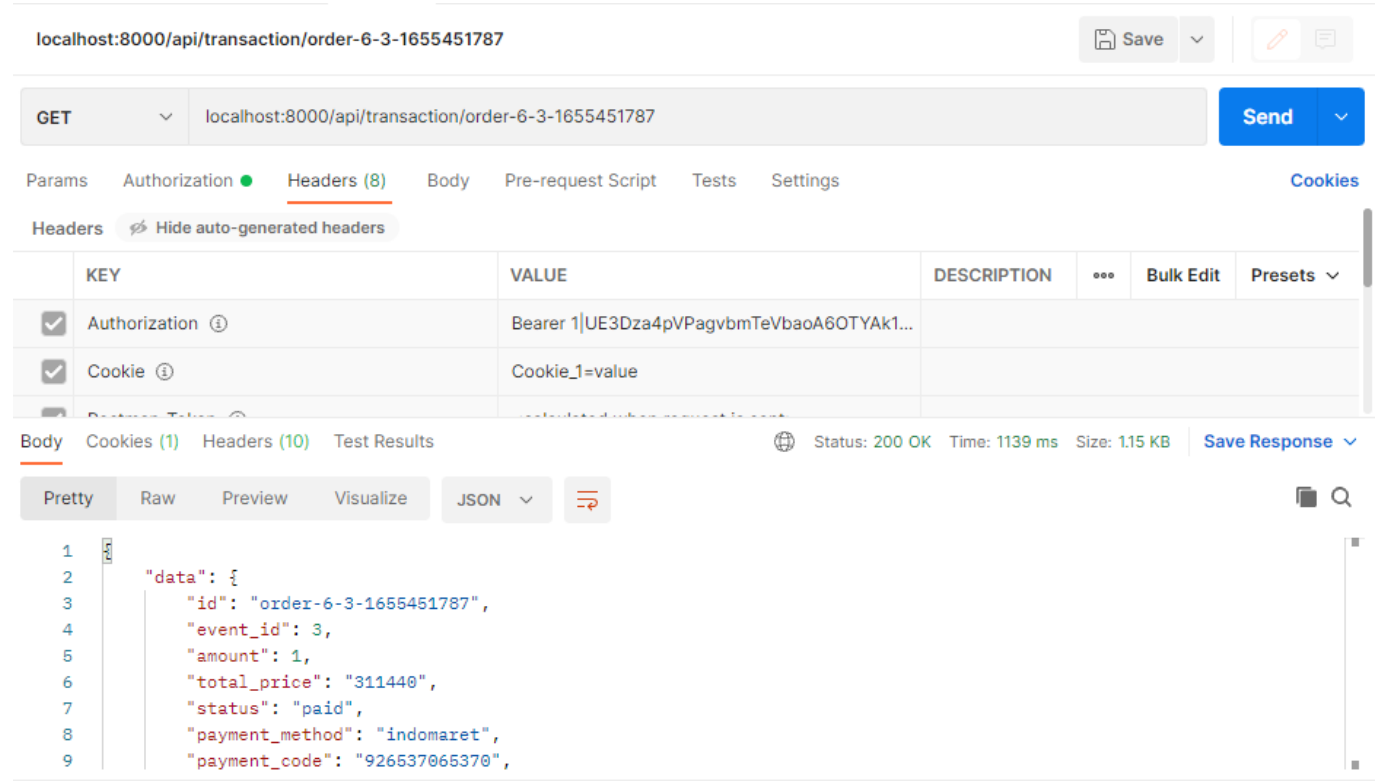
```

{
  "data": {
    "id": "order-6-3-1655451787",
    "event_id": 3,
    "amount": 1,
    "total_price": "311440",
    "status": "paid",
    "payment_method": "indomaret",
    "payment_code": "926537065370",
    "expired_at": "18-06-2022 07:43",
    "ticket_schedule": [
      "11-06-2022"
    ],
    "event": {
      "id": 3,
      "name": "Iste quos pariaturs reprehenderit doloribus voluptas.",
      "description": "Quo repellendus quisquam ut ut voluptatibus impedit. Vitae rem modi quos et sunt eaque. Et quos voluptatem est.",
      "price": "Rp. 311.440",
      "schedule": "[\\"1995-05-23\\",\\"1985-12-20\\"]\\",
      "location": "Reidview",
      "location_description": "Voluptatem doloremque.",
      "rules": "Consequatur voluptates iusto quo rerum molestiae. Dolorum sit libero voluptatem ipsum. Corrupti voluptatem ut dolores fugiat dolorum reiciendis quia.",
      "organization": {
        "id": 1,
        "name": "Kemendikbud RI",
        "slug": "kemendikbud-ri",
        "created_at": "14-06-2022"
      }
    }
  },
  "success": true,
  "total": 6,
  "code": 206
}

```

Example

Berikut adalah contoh response detail transaksi menggunakan aplikasi postman.



Membatalkan transaksi

| Type | Endpoint |
|--------|-------------------------------------|
| DELETE | localhost:8000/api/transaction/{id} |

Headers

| Key | Value |
|---------------|--------------------|
| Authorization | Bearer tokendisini |

Sample Response

Berikut adalah contoh response yang akan didapatkan jika transaksi berhasil dibatalkan.

```
{
  "data": {
    "id": "order-6-3-1655451787",
    "event_id": 3,
    "amount": 1,
    "total_price": "311440",
    "status": "cancelled",
    "payment_method": "indomaret",
    "payment_code": "926537065370",
    "expired_at": "18-06-2022 07:43",
    "ticket_schedule": [
      "11-06-2022"
    ],
  }
}
```

```

    "event": {
      "id": 3,
      "name": "Iste quos pariaturs reprehenderit doloribus voluptas.",
      "description": "Quo repellendus quisquam ut ut voluptatibus impedit. Vitae rem modi quos et sunt eaque. Et quos voluptatem est.",
      "price": "Rp. 311.440",
      "schedule": "\"[\\\"1995-05-23\\\"\",\\\"1985-12-20\\\"]\"",
      "location": "Reidview",
      "location_description": "Voluptatem doloremque.",
      "rules": "Consequatur voluptates iusto quo rerum molestiae. Dolorum sit libero voluptatem ipsum. Corrupti voluptatem ut dolores fugiat dolorum reiciendis quia.",
      "organization": {
        "id": 1,
        "name": "Kemendikbud RI",
        "slug": "kemendikbud-ri",
        "created_at": "14-06-2022"
      }
    },
    "success": true,
    "total": 6,
    "code": 206
  }
}

```

Example

Berikut adalah contoh response membatalkan transaksi menggunakan aplikasi postman.

localhost:8000/api/transaction/order-6-3-1655451787

DELETE localhost:8000/api/transaction/order-6-3-1655451787 Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Headers Hide auto-generated headers

| | KEY | VALUE | DESCRIPTION | ... | Bulk Edit | Presets |
|-------------------------------------|---------------|---|-------------|-----|-----------|---------|
| <input checked="" type="checkbox"/> | Authorization | Bearer 1 UE3Dza4pVPagvbmTeVbaoA6OTYAk1... | | | | |
| <input checked="" type="checkbox"/> | Cookie | Cookie_1=value | | | | |

Body Cookies (1) Headers (10) Test Results

Status: 200 OK Time: 518 ms Size: 1.16 KB Save Response

Pretty Raw Preview Visualize JSON

```

1  {
2    "data": {
3      "id": "order-6-3-1655451787",
4      "event_id": 3,
5      "amount": 1,
6      "total_price": "311440",
7      "status": "cancelled",
8      "payment_method": "indomaret",
9      "payment_code": "926537065370",

```