

# Transaction API

## Documentation

**Version:** 1.1.0 | **Date:** 2026-02-11

**Authentication:** All endpoints require a valid `Bearer Token` in the `Authorization` header.

### 1. Create Transaction

**POST** `/transaction/createTransaction`

**Content-Type:** `multipart/form-data`

#### Description

Creates a new transaction for a customer **owned by the logged-in user**. The server automatically saves the user's ID.

#### Request Body (FormData)

Key	Type	Required	Description
<code>customerId</code>	String	Yes	The <code>_id</code> of the profile. Must belong to the logged-in user.
<code>customerType</code>	String	No	<code>IndividualProfile</code> or <code>CorporateProfile</code> . (Auto-detected if omitted)

Key	Type	Required	Description
invoiceNumber	String	Yes	Unique invoice identifier.
branch	String	Yes	Branch name.
transactionType	String	Yes	Deposit or Withdrawal.
product	String	Yes	Product name.
totalAmount	Number	Yes	Total transaction amount.
payments	String	Yes	JSON String of payment array. Example: [{"mode": "Cash", "amount": 100}]
file	File	No	Optional file attachment.

## Success Response (201 Created)

```
{
  "message": "Transaction created successfully",
  "customerType": "IndividualProfile",
  "transaction": {
    "_id": "653b...",
    "userId": "653a...",
    "customerId": "6539...",
    "invoiceNumber": "INV-001",
    "totalAmount": 150,
    "status": "Success",
    "createdAt": "2023-10-27T10:00:00.000Z"
  }
}
```

## Error Responses

- **400 Bad Request:** Missing required fields.
- **403 Forbidden:** `customerId` does not belong to the logged-in user.
- **404 Not Found:** Customer profile not found.

## 2. Get Transactions

GET

`/transaction/getTransactions`

### Description

Retrieves a paginated list of transactions **belonging to the logged-in user**.

### Query Parameters

Parameter	Type	Default	Description
<code>page</code>	Number	1	Page number.
<code>limit</code>	Number	10	Items per page.
<code>searchType</code>	String	-	Filter by: <code>Name</code> , <code>Core customer number</code> , <code>Mobile</code>
<code>searchValue</code>	String	-	Value to search for.

### Success Response (200 OK)

```
{
  "data": [
    {
      "_id": "653b...",
      "transactionId": "653b...",
      "invoiceNumber": "INV-001",
```

```
        "customerName": "John Doe",
        "totalAmount": 150,
        "status": "Success",
        "date": "2023-10-27T10:00:00.000Z"
      }
    ],
    "pagination": {
      "total": 50,
      "page": 1,
      "limit": 10,
      "pages": 5
    }
  }
}
```

### 3. Cancel Transaction

---

**PATCH**

`/transaction/:transactionId/cancel`

#### Description

---

Cancels a transaction. **User can only cancel their own transactions.**

#### Success Response (200 OK)

---

```
{
  "message": "Transaction cancelled successfully",
  "transaction": {
    "_id": "653b...",
    "status": "Cancelled"
  }
}
```

#### Error Responses

---

- **404 Not Found:** Transaction not found OR does not belong to the user.
- **400 Bad Request:** Transaction is already cancelled.

## 4. Delete Transaction (Soft Delete)

---

**PATCH**

`/transaction/:transactionId/delete`

### Description

---

Soft deletes a transaction. **User can only delete their own transactions.**

### Success Response (200 OK)

---

```
{
  "message": "Transaction deleted successfully",
  "transaction": {
    "_id": "653b...",
    "isDeleted": true
  }
}
```

### Error Responses

---

- **404 Not Found:** Transaction not found OR does not belong to the user.
- **400 Bad Request:** Transaction is already deleted.