

API Documentation

Base URL

/users

1. Register User

Endpoint:

- **POST** /users/register

Request Body:

```
{
  "username": "string",
  "email": "string",
  "password": "string",
  "balance": "float (optional)",
  "role": "string (optional)"
}
```

Response:

- **200 OK:**

```
{
  "success": true,
  "status": 200,
  "data": {
    "id": "UUID",
    "username": "string",
    "email": "string",
    "balance": "float",
    "role": "string",
    "created_at": "datetime"
  }
}
```

- **500 Internal Server Error:**

```
{
  "success": false,
  "status": 500,
}
```

```
"message": "Error during register: {error_message}"
}
```

2. Login User

Endpoint:

- **POST** /users/login

Request Body:

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

Response:

- **200 OK:**

```
{
  "success": true,
  "status": 200,
  "data": {
    "token": "Bearer {jwt_token}"
  }
}
```

- **401 Unauthorized:**

```
{
  "success": false,
  "status": 401,
  "message": "Invalid credentials"
}
```

- **400 Bad Request:**

```
{
  "success": false,
  "status": 400,
  "message": "Invalid credentials"
}
```

3. Get User Profile

Endpoint:

- `GET /users/{id}`

Path Parameters:

- `id` (UUID): User ID

Response:

- **200 OK:**

```
{
  "success": true,
  "status": 200,
  "data": {
    "user": {
      "id": "UUID",
      "username": "string",
      "email": "string",
      "balance": "float",
      "role": "string",
      "created_at": "datetime"
    }
  }
}
```

- **404 Not Found:**

```
{
  "success": false,
  "status": 404,
  "message": "User not found"
}
```

4. Update User Profile

Endpoint:

- `PUT /users/{id}`

Path Parameters:

- `id` (UUID): User ID

Request Body:

```
{
  "username": "string (optional)",
  "email": "string (optional)",
  "role": "string (optional)"
}
```

Response:

- **200 OK:**

```
{
  "success": true,
  "status": 200,
  "message": "Profile updated successfully"
}
```

- **401 Unauthorized:**

```
{
  "success": false,
  "status": 401,
  "message": "Unauthorised request!"
}
```

5. Update User Password

Endpoint:

- **PATCH** /users/{id}/reset_password

Path Parameters:

- **id** (UUID): User ID

Request Body:

```
{
  "current_password": "string",
  "new_password": "string"
}
```

Response:

- **202 Accepted:**

```
{
  "success": true,
  "status": 202,
  "message": "Password update successful"
}
```

- **401 Unauthorized:**

```
{
  "success": false,
  "status": 401,
  "message": "Current password is incorrect"
}
```

- **404 Not Found:**

```
{
  "success": false,
  "status": 404,
  "message": "User not found"
}
```

6. Update User Balance

Endpoint:

- **PUT** /users/{id}/update_balance

Path Parameters:

- **id** (UUID): User ID

Request Body:

```
{
  "amount": "float",
  "transaction_type": "Withdrawal | Deposit"
}
```

Response:

- **200 OK:**

```
{
  "success": true,
  "status": 200,
  "message": "User Balance Updated"
}
```

- **401 Unauthorized:**

```
{
  "success": false,
  "status": 401,
  "message": "Unauthorised request!"
}
```

- **404 Not Found:**

```
{
  "success": false,
  "status": 404,
  "message": "User not found"
}
```

7. Delete User

Endpoint:

- **DELETE** /users/{id}

Path Parameters:

- **id** (UUID): User ID

Response:

- **200 OK:**

```
{
  "success": true,
  "status": 200,
  "message": "User deleted successfully!"
}
```

- **401 Unauthorized:**

```
{
  "success": false,
  "status": 401,
  "message": "Only user and admin can delete!"
}
```

- **500 Internal Server Error:**

```
{
  "success": false,
  "status": 500,
  "message": "Failed to get database connection"
}
```

Base URL

/transactions

1. Create Transaction

Endpoint:

- **POST** /transactions/transact

Request Body:

```
{
  "sender_id": "UUID",
  "recipient_id": "UUID",
  "amount": "float",
  "description": "string"
}
```

Response:

- **200 OK:**

```
{
  "success": true,
  "status": 200,
  "message": "Transaction Successful",
  "data": {
    "id": "UUID",
    "sender_id": "UUID",

```

```
"recipient_id": "UUID",  
"amount": "float",  
"description": "string",  
"created_at": "datetime"  
}  
}
```

- **400 Bad Request:**

```
{  
  "success": false,  
  "status": 400,  
  "message": "Only can Transfer from own account"  
}
```

```
{  
  "success": false,  
  "status": 400,  
  "message": "Insuffient Balance"  
}
```

- **404 Not Found:**

```
{  
  "success": false,  
  "status": 404,  
  "message": "Not Found"  
}
```

- **500 Internal Server Error:**

```
{  
  "success": false,  
  "status": 500,  
  "message": "Something went wrong!"  
}
```

2. Get Transaction By ID

Endpoint:

- **GET** /transactions/{id}

Path Parameters:

- **id** (UUID): Transaction ID

Response:

- **200 OK:**

```
{
  "success": true,
  "status": 200,
  "data": {
    "id": "UUID",
    "sender_id": "UUID",
    "recipient_id": "UUID",
    "amount": "float",
    "sender_username": "string",
    "sender_email": "string",
    "recipient_username": "string",
    "recipient_email": "string"
  }
}
```

- **404 Not Found:**

```
{
  "success": false,
  "status": 404,
  "message": "Transaction Not Found"
}
```

3. List Transactions by User

Endpoint:

- **GET** /transactions/user/{user_id}

Path Parameters:

- **user_id** (UUID): User ID

Response:

- **200 OK:**

```
{
  "success": true,
  "status": 200,
```

```
"data": [  
  {  
    "id": "UUID",  
    "sender_id": "UUID",  
    "recipient_id": "UUID",  
    "amount": "float",  
    "description": "string",  
    "created_at": "datetime"  
  },  
  ...  
]  
}
```

- **400 Bad Request:**

```
{  
  "success": false,  
  "status": 400,  
  "message": "Only can see own transaction!"  
}
```

- **500 Internal Server Error:**

```
{  
  "success": false,  
  "status": 500,  
  "message": "Something went wrong!"  
}
```

Notes:

- All endpoints require authentication.
- Errors return a standard error message format with **success**, **status**, and **message** fields.