

MoneyRAPI

Rauan Caracciolo

1. GET /transactions/paginated?page={n}

Descrição: Retorna uma lista paginada com todas as transações cadastradas no sistema.

- **Método HTTP:** GET
 - **Endpoint:** /transactions/paginated?page=0
 - **Parâmetros:**
 - page (obrigatório): número da página (ex: 0, 1, 2...).
 - **Códigos de resposta:**
 - 200 OK – Lista retornada com sucesso.
 - 500 Internal Server Error – Erro ao buscar os dados.
-

2. GET /transactions/category/{categoria}

Descrição: Retorna transações filtradas por categoria.

- **Método HTTP:** GET
 - **Endpoint:** /transactions/category/{categoria}
 - **Parâmetros:**
 - categoria (obrigatório): nome da categoria.
 - **Códigos de resposta:**
 - 200 OK – Lista retornada com sucesso.
 - 404 Not Found – Nenhuma transação encontrada.
 - 500 Internal Server Error – Erro ao buscar os dados.
-

3. GET /transactions/type/{tipo}

Descrição: Retorna transações filtradas por tipo (income ou expense).

- **Método HTTP:** GET
- **Endpoint:** /transactions/type/{tipo}

- **Parâmetros:**
 - tipo (obrigatório): income ou expense.
 - **Códigos de resposta:**
 - 200 OK – Lista retornada com sucesso.
 - 400 Bad Request – Tipo inválido.
 - 500 Internal Server Error – Erro ao buscar os dados.
-

4. GET /transactions/categoryandtype/{categoria}+{tipo}

Descrição: Retorna transações filtradas por categoria e tipo simultaneamente.

- **Método HTTP:** GET
 - **Endpoint:** /transactions/categoryandtype/{categoria}+{tipo}
 - **Códigos de resposta:**
 - 200 OK – Lista retornada com sucesso.
 - 404 Not Found – Nenhuma transação encontrada.
 - 500 Internal Server Error – Erro ao buscar os dados.
-

5. GET /transactions/relatory

Descrição: Retorna um resumo financeiro com totais e agrupamentos por categoria.

- **Método HTTP:** GET
- **Endpoint:** /transactions/relatory
- **Resposta esperada (JSON):**

```
{
  "totalIncome": 552.2,
  "totalExpense": 11250.43,
  "actualMoney": -10698.23,
  "categoryIncome": {
    "Grocery": 200.0,
    "job": 200.0,
    "Teste": 32.2
  },
  "categoryExpense": {
    "Carro": 11200.43,
    "Mercado": 50.0
  }
}
```

- **Códigos de resposta:**

- 200 OK – Relatório gerado com sucesso.
 - 500 Internal Server Error – Erro ao gerar o relatório.
-

6. GET /transactions/{id}

Descrição: Retorna os dados de uma transação específica.

- **Método HTTP:** GET
 - **Endpoint:** /transactions/{id}
 - **Parâmetros:**
 - id (obrigatório): identificador da transação.
 - **Códigos de resposta:**
 - 200 OK – Transação encontrada.
 - 404 Not Found – Transação não existe.
 - 500 Internal Server Error – Erro interno.
-

7. POST /transactions

Descrição: Cadastra uma nova transação.

- **Método HTTP:** POST
- **Endpoint:** /transactions
- **Corpo da requisição (JSON):**

```
{  
  "value": 150.0,  
  "desc": "Salário",  
  "type": "income",  
  "cat": "job",  
  "time": "10-07-2025"  
}
```

- **Códigos de resposta:**
 - 201 Created – Transação cadastrada com sucesso.
 - 400 Bad Request – Dados inválidos ou ausentes.
 - 500 Internal Server Error – Erro ao cadastrar.
-

8. PUT /transactions

Descrição: Atualiza os dados de uma transação existente.

- **Método HTTP:** PUT
- **Endpoint:** /transactions
- **Corpo da requisição (JSON):**

```
{
  "id": 1,
  "value": 180.0,
  "desc": "Salário atualizado",
  "type": "income",
  "cat": "job",
  "time": "11-07-2025"
}
```

Códigos de resposta:

- 200 OK – Transação atualizada.
- 400 Bad Request – Dados inválidos.
- 404 Not Found – Transação não encontrada.
- 500 Internal Server Error – Erro ao atualizar.

9. DELETE /transactions/{id}

Descrição: Remove uma transação existente do sistema.

- **Método HTTP:** DELETE
- **Endpoint:** /transactions/{id}
- **Parâmetros:**
 - id (obrigatório): identificador da transação.
- **Códigos de resposta:**
 - 200 OK – Transação excluída.
 - 404 Not Found – Transação não existe.
 - 500 Internal Server Error – Erro ao excluir.
-