Expense Tracker API Contract

KOTLA ANUDEEP REDDY

August 11, 2025

Contents

1 Overview

This document defines the REST API endpoints, request and response formats, and authentication methods for the Expense Tracker application.

2 Authentication

2.1 Register User

- **POST** /api/auth/register
- **Request Body:**

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

- **Response:** 201 Created with user profile (without password)

2.2 Login

- **POST** /api/auth/login
- **Request Body:**

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

- **Response:**

```
{
  "access_token": "jwt_token",
  "refresh_token": "refresh_token"
}
```

2.3 Refresh Token

- **POST** /api/auth/refresh
- **Request Body:**

```
{
  "refresh_token": "string"
}
```

- **Response:**

```
{
  "access_token": "jwt_token"
}
```

2.4 Logout

- **POST** /api/auth/logout
- **Headers:** Authorization: Bearer <access_token>
- **Request Body:**

```
{
  "refresh_token": "string"
}
```

- **Response:** 204 No Content

3 User Profile

3.1 Get Profile

- **GET** /api/user/profile
- **Headers:** Authorization: Bearer <access_token>
- **Response:**

```
{
  "user_id": "uuid",
  "name": "string",
  "email": "string",
  "created_at": "timestamp",
  "updated_at": "timestamp"
}
```

3.2 Update Profile

- **PUT** /api/user/profile
- **Headers:** Authorization: Bearer <access_token>
- **Request Body:**

```
{
  "name": "string"
}
```

- **Response:** Updated user profile object

4 Bank Accounts

4.1 Add Bank Account

- **POST** /api/bank-accounts
- **Headers:** Authorization: Bearer <access_token>
- **Request Body:**

```
{
  "bank_name": "string",
  "account_mask": "string"
}
```

- **Response:** Created bank account object

4.2 List Bank Accounts

- **GET** /api/bank-accounts
- **Headers:** Authorization: Bearer <access_token>
- **Response:** List of bank account objects

4.3 Delete Bank Account

- **DELETE** /api/bank-accounts/{account_id}
- **Headers:** Authorization: Bearer <access_token>
- **Response:** 204 No Content

5 Transactions

5.1 Add Transaction

- **POST** /api/transactions
- **Headers:** Authorization: Bearer <access_token>
- **Request Body:**

```
{
  "account_id": "uuid (optional)",
  "amount": 100.50,
  "category_id": "uuid (optional)",
  "upi_app": "string (optional)",
  "location": "string (optional)",
  "mcc": "string (optional)",
  "timestamp": "ISO8601 timestamp",
  "notes": "string (optional)"
}
```

- **Response:** Created transaction object

5.2 List Transactions

- **GET** /api/transactions?start_date={date}end_date={date}category_id={uuid}limit={int}offset
- **Headers:** Authorization: Bearer <access_token>
- **Response:** Paginated list of transactions

5.3 Update Transaction

- **PUT** /api/transactions/{transaction_id}
- **Headers:** Authorization: Bearer <access_token>
- **Request Body:** Same fields as Add Transaction (all optional except amount)
- **Response:** Updated transaction object

5.4 Delete Transaction

- **DELETE** /api/transactions/{transaction_id}
- **Headers:** Authorization: Bearer <access_token>
- **Response:** 204 No Content

6 Categories

6.1 List Categories

- **GET** /api/categories
- **Headers:** Authorization: Bearer <access_token>
- **Response:** List of system and user categories

7 Budgets

7.1 Create Budget

- **POST** /api/budgets
- **Headers:** Authorization: Bearer <access_token>
- **Request Body:**

```
{
  "category_id": "uuid",
  "limit_amount": 5000,
  "start_date": "YYYY-MM-DD",
  "end_date": "YYYY-MM-DD"
}
```

- **Response:** Created budget object

7.2 List Budgets

- **GET** /api/budgets
- **Headers:** Authorization: Bearer <access_token>
- **Response:** List of budgets

7.3 Update Budget

- **PUT** /api/budgets/{budget_id}
- **Headers:** Authorization: Bearer <access_token>
- **Request Body:** Same as Create Budget (all fields optional)
- **Response:** Updated budget object

7.4 Delete Budget

- **DELETE** /api/budgets/{budget_id}
- **Headers:** Authorization: Bearer <access_token>
- **Response:** 204 No Content

8 Alerts

8.1 Get Alerts

- **GET** /api/alerts
- **Headers:** Authorization: Bearer <access_token>
- **Response:** List of alerts (unread/read)

8.2 Mark Alert as Read

- **POST** /api/alerts/{alert_id}/read
- **Headers:** Authorization: Bearer <access_token>
- **Response:** Updated alert object

9 Allocations

9.1 Get Allocations

- **GET** /api/allocations
- **Headers:** Authorization: Bearer <access_token>
- **Response:** List of allocation pools (Fixed, Spend, Save) with current amounts

9.2 Update Allocation

- **PUT** /api/allocations/{allocation_id}
- **Headers:** Authorization: Bearer <access_token>
- **Request Body:**

```
{
  "target_pct": 30,
  "target_amount": 6000
}
```

- **Response:** Updated allocation object

10 Audit Logs

10.1 Get Audit Logs

- **GET** /api/audit-logs?limit={int}offset={int}
- **Headers:** Authorization: Bearer <access_token>
- **Response:** List of audit logs (admin only or user-specific)

11 Push Notifications

11.1 Register FCM Token

- **POST** /api/fcm-tokens
- **Headers:** Authorization: Bearer <access_token>
- **Request Body:**

```
{
  "token": "string",
  "device_info": "string (optional)"
}
```

- **Response:** Created FCM token object

11.2 Delete FCM Token

- **DELETE** /api/fcm-tokens/{fcm$_token_id$}**Headers:***Authorization* : *Bearer<access_token>*

- **Response:** 204 No Content

12 Error Handling

All API responses on error use standard HTTP codes and return JSON:

```
{
  "error": "string",
  "message": "string",
  "details": {}
}
```

13 Security

- All endpoints except registration and login require JWT bearer authentication.
- Passwords must be stored hashed using a strong algorithm.
- Refresh tokens are stored securely and can be revoked.
- Rate limiting and logging should be implemented to prevent abuse.