# Expense Tracker Database Schema

KOTLA ANUDEEP REDDY

August 11, 2025

## Contents

# 1 Overview

This document describes the database schema for the Expense Tracker application. It includes all tables, columns, data types, primary keys (PK), foreign keys (FK), and constraints.

# 2 Tables

## 2.1 User

Stores registered users and basic authentication data.

| Column | Data Type | Constraints | Description |
|---|---|---|---|
| user_id | UUID | PK, Not Null | Unique user identifier |
| name | VARCHAR(100) | Not Null | User's full name |
| email | VARCHAR(150) | Not Null, Unique | User's email address |
| password_hash | VARCHAR(256) | Not Null | Hashed password |
| created_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Account creation time |
| updated_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Last profile update time |

## 2.2 Refresh_Tokens

Supports multiple refresh tokens per user (for multi-device sessions).

| | | | |
|---|---|---|---|
| token_id | UUID | PK, Not Null | Unique token identifier |
| user_id | UUID | FK(User.user_id), Not Null | Owner user ID |
| token | TEXT | Not Null | Refresh token string |
| expires_at | TIMESTAMP | Not Null | Token expiry timestamp |
| revoked | BOOLEAN | Not Null, Default FALSE | Revocation status |
| created_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Creation time |

## 2.3 FCM_Tokens

Stores push notification tokens per device.

| | | | |
|---|---|---|---|
| fcm_token_id | UUID | PK, Not Null | Unique token identifier |
| user_id | UUID | FK(User.user_id), Not Null | Owner user |
| token | TEXT | Not Null | Firebase Cloud Messaging token |
| device_info | VARCHAR(255) | Nullable | Optional device description |
| created_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Creation time |

## 2.4 Bank_Accounts

Linked bank accounts via Account Aggregator.

| | | | |
|---|---|---|---|
| account_id | UUID | PK, Not Null | Unique bank account ID |
| user_id | UUID | FK(User.user_id), Not Null | Owner user |
| bank_name | VARCHAR(100) | Not Null | Bank name (e.g., SBI) |
| account_mask | VARCHAR(20) | Not Null | Masked account number (e.g., XXXX1234) |
| last_sync | TIMESTAMP | Nullable | Last synchronization time |
| created_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Linked date |
| updated_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Last update |

## 2.5 Category

Transaction categories.

| | | | |
|---|---|---|---|
| category_id | UUID | PK, Not Null | Unique category ID |
| name | VARCHAR(50) | Not Null | Category name (e.g., Food, Rent) |
| color_code | VARCHAR(7) | Nullable | Hex color code for UI |
| created_by | VARCHAR(20) | Not Null | 'system' or 'user' |

## 2.6 Transaction

User transactions with UPI app, location, MCC info.

| | | | |
|---|---|---|---|
| transaction_id | UUID | PK, Not Null | Unique transaction ID |
| user_id | UUID | FK(User.user_id), Not Null | Owner user |
| account_id | UUID | FK(Bank_Account.account_id), Nullable | Linked bank account |
| amount | DECIMAL(12,2) | Not Null | Transaction amount |
| category_id | UUID | FK(Category.category_id), Nullable | Category assigned |
| upi_app | VARCHAR(50) | Nullable | UPI app used (PhonePe, Paytm, etc.) |
| location | VARCHAR(255) | Nullable | Location data (lat/long or address) |
| mcc | VARCHAR(10) | Nullable | Merchant Category Code |
| timestamp | TIMESTAMP | Not Null | Transaction date/time |
| notes | TEXT | Nullable | User notes |
| created_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Record creation time |
| updated_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Last update time |

## 2.7 Budget

Spending limits per user and category.

| | | | |
|---|---|---|---|
| budget_id | UUID | PK, Not Null | Unique budget ID |
| user_id | UUID | FK(User.user_id), Not Null | Owner user |
| category_id | UUID | FK(Category.category_id), Not Null | Category |
| limit_amount | DECIMAL(12,2) | Not Null | Spending limit |
| start_date | DATE | Not Null | Budget start date |
| end_date | DATE | Not Null | Budget end date |
| created_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Creation time |
| updated_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Last update time |

## 2.8 Alert

User notifications.

| | | | |
|---|---|---|---|
| alert_id | UUID | PK, Not Null | Unique alert ID |
| user_id | UUID | FK(User.user_id), Not Null | Recipient user |
| type | VARCHAR(50) | Not Null | Alert type (limit reached, suspicious txn) |
| message | TEXT | Not Null | Alert message |
| timestamp | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Created time |
| status | VARCHAR(10) | Not Null, Default 'unread' | Alert status |

## 2.9 Allocation

Money allocation pools (Fixed, Spend, Save).

| | | | |
|---|---|---|---|
| allocation_id | UUID | PK, Not Null | Unique allocation ID |
| user_id | UUID | FK(User.user_id), Not Null | Owner user |
| name | VARCHAR(50) | Not Null | Allocation name (Fixed, Spend, Save) |
| target_pct | INT | Nullable | Target percentage of total income |
| target_amount | DECIMAL(12,2) | Nullable | Target fixed amount |
| current_amount | DECIMAL(12,2) | Not Null, Default 0.00 | Current balance |
| last_funded | TIMESTAMP | Nullable | Last fund update time |
| created_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Creation time |
| updated_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Last update time |

## 2.10 Audit_Logs

Tracks user and system actions for security and debugging.

| | | | |
|---|---|---|---|
| log_id | UUID | PK, Not Null | Unique log entry ID |
| user_id | UUID | FK(User.user_id), Nullable | User performing action |
| action | VARCHAR(100) | Not Null | Action description (login, txn_update, etc.) |
| details | TEXT | Nullable | Additional details (JSON/text) |
| ip_address | VARCHAR(45) | Nullable | IP address of user/device |
| created_at | TIMESTAMP | Not Null, Default CURRENT_TIMESTAMP | Log timestamp |

# 3 Indexes and Constraints

- Primary keys on all `id` columns.
- Foreign key constraints enforce referential integrity.
- Unique constraint on `User.email`.
- Indexes on `Transaction.user_id`, `Transaction.category_id`, `Budget.user_id`, `Budget.category_id`.
- Index on `Bank_Accounts.user_id` for fast lookups.

# 4 Notes

- Passwords must be stored hashed securely (e.g., bcrypt).
- Refresh tokens and FCM tokens are stored in separate tables to support multiple sessions and devices.
- Location can be stored as a simple string or structured JSON, depending on implementation.
- MCC codes are optional and used for advanced merchant categorization.
- Audit logs provide a security trail and can be used for debugging.