For a small-scale personal expense tracker project, you can use a simple relational database with a few tables. Here's a minimal database schema using SQL:

1. **User Table:**
   * Store information about registered users.
   * Fields: **user\_id** (primary key), **username**, **password**, **email**, and any other relevant user information.

sqlCopy code

CREATE TABLE users ( user\_id INT AUTO\_INCREMENT PRIMARY KEY, username VARCHAR(50) NOT NULL, password VARCHAR(255) NOT NULL, email VARCHAR(100) NOT NULL );

1. **Expense Table:**
   * Store details of the expenses entered by users.
   * Fields: **expense\_id** (primary key), **user\_id** (foreign key), **amount**, **date**, **category\_id** (foreign key), and **description**.

sqlCopy code

CREATE TABLE expenses ( expense\_id INT AUTO\_INCREMENT PRIMARY KEY, user\_id INT, amount DECIMAL(10, 2) NOT NULL, date DATE NOT NULL, category\_id INT, description TEXT, FOREIGN KEY (user\_id) REFERENCES users(user\_id), FOREIGN KEY (category\_id) REFERENCES expense\_categories(category\_id) );

1. **Expense Categories Table:**
   * Store predefined expense categories.
   * Fields: **category\_id** (primary key) and **category\_name**.

sqlCopy code

CREATE TABLE expense\_categories ( category\_id INT AUTO\_INCREMENT PRIMARY KEY, category\_name VARCHAR(50) NOT NULL );

With this simple database schema, you can manage user registration and authentication, record expenses with their categories, and list the entered expenses. You may also need to implement a database connection in your web application code, and set up the appropriate queries to interact with this database. Depending on your programming language and framework, you can create models or classes to handle these database operations.