**Expense Tracking Project**

**Inputs:**

1. **User Information:**
   * User ID
   * User Name
   * Email Address
2. **Expense Entries:**
   * Date of Expense
   * Amount Spent
   * Category (e.g., Food, Transportation, Utilities, Entertainment)
   * Description (optional)
   * Payment Method (e.g., Cash, Credit Card, Bank Transfer)
3. **Recurring Expenses:**
   * Frequency (e.g., Daily, Weekly, Monthly)
   * Amount
   * Category
   * Start Date
   * End Date (if applicable)
4. **Budget Settings:**
   * Monthly Budget Limit
   * Category-Specific Budget Limits
5. **Reports Request:**
   * Date Range for Report
   * Categories to Include/Exclude
   * Type of Report (e.g., Summary, Detailed)

**Outputs:**

1. **Expense Summary:**
   * Total Expenses for a given period
   * Breakdown by Category
   * Comparison to Budget
2. **Detailed Expense Report:**
   * List of all expenses within a specified date range
   * Individual entries with date, amount, category, and description
3. **Recurring Expense Overview:**
   * List of all recurring expenses
   * Next due date for each recurring expense
4. **Budget Status:**
   * Current spending vs. budget limit
   * Alerts for overspending in specific categories
5. **Visual Reports:**
   * Graphs/Charts showing spending trends over time
   * Pie charts for category-wise spending distribution

Sql code chatgpt:

-- 1. Users Table

CREATE TABLE users (

user\_id NUMBER PRIMARY KEY,

user\_name VARCHAR2(100) NOT NULL,

email VARCHAR2(150) UNIQUE NOT NULL

);

-- 2. Expenses Table

CREATE TABLE expenses (

expense\_id NUMBER PRIMARY KEY,

user\_id NUMBER NOT NULL,

expense\_date DATE NOT NULL,

amount NUMBER(10,2) NOT NULL,

category VARCHAR2(50) NOT NULL,

description VARCHAR2(255),

payment\_method VARCHAR2(50) NOT NULL,

FOREIGN KEY (user\_id) REFERENCES users(user\_id) ON DELETE CASCADE

);

-- 3. Recurring Expenses Table

CREATE TABLE recurring\_expenses (

recurring\_id NUMBER PRIMARY KEY,

user\_id NUMBER NOT NULL,

frequency VARCHAR2(20) NOT NULL, -- e.g., Daily, Weekly, Monthly

amount NUMBER(10,2) NOT NULL,

category VARCHAR2(50) NOT NULL,

start\_date DATE NOT NULL,

end\_date DATE, -- Can be NULL for indefinite recurring expenses

FOREIGN KEY (user\_id) REFERENCES users(user\_id) ON DELETE CASCADE

);

-- 4. Budget Settings Table

CREATE TABLE budget\_settings (

budget\_id NUMBER PRIMARY KEY,

user\_id NUMBER NOT NULL,

monthly\_budget\_limit NUMBER(10,2) NOT NULL,

category VARCHAR2(50), -- NULL means overall budget, else category-specific

category\_budget\_limit NUMBER(10,2),

FOREIGN KEY (user\_id) REFERENCES users(user\_id) ON DELETE CASCADE

);

-- 5. Reports Table (For Requesting Reports)

CREATE TABLE reports (

report\_id NUMBER PRIMARY KEY,

user\_id NUMBER NOT NULL,

start\_date DATE NOT NULL,

end\_date DATE NOT NULL,

included\_categories VARCHAR2(255), -- Comma-separated values e.g., 'Food,Entertainment'

report\_type VARCHAR2(20) NOT NULL, -- Summary, Detailed

FOREIGN KEY (user\_id) REFERENCES users(user\_id) ON DELETE CASCADE

);

Sql code :

////////it will create a table named”users\_expense”;

create table users\_expense(expense\_id number(10) not null,expense\_name varchar2(100) not null,expense\_email varchar2(150) unique not null);

/////it alter users\_expense

ALTER TABLE users\_expense ADD CONSTRAINT pk\_users\_expense PRIMARY KEY (expense\_id);

/////Describe:

Name Null? Type

----------------------------------------- -------- ----------------------------

EXPENSE\_ID NOT NULL NUMBER(10)

EXOENSE\_NAME NOT NULL VARCHAR2(100)

EXPENSE\_EMAIL NOT NULL VARCHAR2(150)

//////////IT WILL CRATE A TABEL NAMED”expenses\_expense”;

CREATE TABLE expenses\_expense (

expense\_id NUMBER PRIMARY KEY,

user\_expense\_id NUMBER(10) NOT NULL,

expense\_date DATE NOT NULL,

expense\_amount NUMBER(10,2) NOT NULL,

expense\_category VARCHAR2(50) NOT NULL,

expense\_description VARCHAR2(255),

expense\_payment\_method VARCHAR2(50) NOT NULL,

CONSTRAINT fk\_user FOREIGN KEY (user\_expense\_id) REFERENCES users\_expense(expense\_id) ON DELETE CASCADE

);

////////describe expenses\_expense

Name Null? Type

----------------------------------------- -------- ----------------------------

EXPENSE\_ID NOT NULL NUMBER

USER\_EXPENSE\_ID NOT NULL NUMBER(10)

EXPENSE\_DATE NOT NULL DATE

EXPENSE\_AMOUNT NOT NULL NUMBER(10,2)

EXPENSE\_CATEGORY NOT NULL VARCHAR2(50)

EXPENSE\_DESCRIPTION VARCHAR2(255)

EXPENSE\_PAYMENT\_METHOD NOT NULL VARCHAR2(50)

//it will create table named “reccuring\_expenses”

SQL> CREATE TABLE recurring\_expenses (

recurring\_id NUMBER PRIMARY KEY,

user\_id NUMBER NOT NULL,

frequency VARCHAR2(20) NOT NULL,

amount NUMBER(10,2) NOT NULL,

category VARCHAR2(50) NOT NULL,

start\_date DATE NOT NULL,

end\_date DATE,

CONSTRAINT fk\_recurring\_user FOREIGN KEY (user\_id) REFERENCES users\_expense(expense\_id) ON DELETE CASCADE

);

////// Describe reccuring\_expenses

Name Null? Type

----------------------------------------- -------- ----------------------------

RECURRING\_ID NOT NULL NUMBER

USER\_ID NOT NULL NUMBER

FREQUENCY NOT NULL VARCHAR2(20)

AMOUNT NOT NULL NUMBER(10,2)

CATEGORY NOT NULL VARCHAR2(50)

START\_DATE NOT NULL DATE

END\_DATE DATE

/////sequence creation for expense\_id

DROP SEQUENCE users\_expense\_seq;

CREATE SEQUENCE users\_expense\_seq

START WITH 1

INCREMENT BY 1

NOCACHE;