Personal Finance Tracker

The goal of the project is to create a web app that allows users to strategically plan their spending for each month. It gives the user a smart way to follow their finances for a extended period or specified period with a dashboard that gives the user the statistics of their spending and shows where they have spent too much etc. The MVP will contain only simple features whereas later version will contain AI agents that give insight into the users spending and offers advice, a smart chatbot.

The MVP

List of functions:
User can set monthly income for a selected month.

User can create/edit/delete categories (e.g., Rent, Gym, Food).

User can set a monthly budget per category (default repeats every month, editable per month).

User can add/edit/delete transactions:

- amount, date, category, description
- type: expense or income

User can view transaction history:

- all transactions for a month
- filter by category

Dashboard for selected time period shows:

- total income, total expenses, balance
- spending by category
- budget vs actual per category

User can choose a time period (from–to) and see totals + category breakdown.

Data model draft

User

- id
- name
- email
- password_hash

- country
- currency

Income

- id
- user_id
- month
- amount

Category

- id
- user_id
- name
- default_budget
- description
- active

Transaction

- id
- user_id
- category_id
- type (expense | income)
- amount
- date
- description
- created_at

Backend architecture

Models/

- User
- Income
- Category
- Transaction

Schemas/

- User: in, out, update
- Income: in, out, update
- Category: in, out, update
- Transaction: in, out, update

Repositories/

- User
- Income
- Category
- Transaction

Services/

Api/routes/

- Get, post, update, delete for all

API contracts

POST /categories

- Req:
  - {
  - "name": "Rent",
  - "default_budget": 600,
  - "description": "Monthly apartment rent",
  - "active": true
  - }
- Res:
  - {
  - "id": auto,
  - "name": "Rent",
  - "default_budget": 600,
  - "description": "Monthly apartment rent",
  - "active": true
  - }

GET /categories

- Res:
  - [
  - {
  - "id": auto,
  - "name": "Rent",
  - "default_budget": 600,
  - "description": "Monthly apartment rent",
  - "active": true
  - }
  -]

PUT /income/{month}

- Req:
  - {
  - "amount": 230
  - }
- Res:
  - {
  - "amount": 230,
  - "month": "2026-01"
  - }

GET /income/{month}

- Res:
  - {
  - "amount": 230,
  - "month": "2026-01"
  - }

POST /transactions

- Req:
  - {
  - "category_id": 2,
  - "type": "expense",
  - "description": "I bought a ball",
  - "amount": 20,
  - "date": "2026-01-15"
  - }
- Res:
  - [
  - {
  - "id": 2,
  - "category_id": 2,
  - "type": "expense",
  - "description": "I bought a ball",
  - "amount": 20,
  - "date": "2026-01-15",
  - "created_at": "2026-01-15T18:22:11Z"
  - }
  -]

GET /transactions ?month=YYYY-MM

- Res
  - [
  - {
  - "id": 2,
  - "category_id":2,
  - "type": "expense",
  - "description": "I bought a ball",
  - "amount": 20,
  - "created_at": "2026-01-15T18:22:11Z"
  - }
  -]

DELETE /transactions/{id}

DELETE /categories/{id}

GET /summary?from=YYYY-MM&to=YYYY-MM

Notice: For MVP, the transactions.items list is limited (e.g. last 100–200 transactions) to avoid large payloads.

- res
  - {
  - "period": {
  - "from": "2026-01",
  - "to": "2026-03",
  - "months": ["2026-01", "2026-02", "2026-03"]
  - },

  - "totals": {
  - "income": 7500.00,
  - "expenses": 4200.50,
  - "balance": 3299.50,
  - "savings_rate": 0.44
  - },

  - "monthly": [

- {
- "month": "2026-01",
- "income": 2500.00,
- "expenses": 1400.50,
- "balance": 1099.50,
- "savings_rate": 0.44
- },
- {
- "month": "2026-02",
- "income": 2500.00,
- "expenses": 1500.00,
- "balance": 1000.00,
- "savings_rate": 0.40
- },
- {
- "month": "2026-03",
- "income": 2500.00,
- "expenses": 1300.00,
- "balance": 1200.00,
- "savings_rate": 0.48
- }
-],

- "categories": [
- {
- "category_id": "uuid-rent",
- "name": "Rent",
- "active": true,

- "planned": 600.00,
- "actual": 600.00,
- "remaining": 0.00,
- "utilization": 1.00,

- "monthly_breakdown": [
- { "month": "2026-01", "planned": 600.00, "actual": 600.00, "remaining": 0.00 },

- { "month": "2026-02", "planned": 600.00, "actual": 600.00, "remaining": 0.00 },
- { "month": "2026-03", "planned": 600.00, "actual": 600.00, "remaining": 0.00 }
-]
- },
- {
- "category_id": "uuid-food",
- "name": "Food",
- "active": true,

- "planned": 900.00,
- "actual": 1200.50,
- "remaining": -300.50,
- "utilization": 1.33,

- "monthly_breakdown": [
- { "month": "2026-01", "planned": 300.00, "actual": 420.50, "remaining": -120.50 },
- { "month": "2026-02", "planned": 300.00, "actual": 450.00, "remaining": -150.00 },
- { "month": "2026-03", "planned": 300.00, "actual": 330.00, "remaining": -30.00 }
-]
- }
-],

- "transactions": {
- "count": 42,
- "items": [
- {
- "id": "uuid-t1",
- "date": "2026-01-15",
- "type": "expense",
- "amount": 35.50,
- "category_id": "uuid-food",
- "category_name": "Food",
- "description": "Groceries",

- "created_at": "2026-01-15T18:22:11Z"
- }
-]
- },

- "insights": {
- "over_budget_categories": [
- {
- "category_id": "uuid-food",
- "name": "Food",
- "planned": 900.00,
- "actual": 1200.50,
- "over_by": 300.50
- }
-],
- "top_categories_by_spend": [
- { "category_id": "uuid-rent", "name": "Rent", "amount": 1800.00 },
- { "category_id": "uuid-food", "name": "Food", "amount": 1200.50 }
-],
- "notes": [
- "Food spending is above budget in this period."
-]
- }
- }