

HematWeb – Web Application Pencatat Keuangan Personal

I. Project Overview

What is HematWeb?

This is an app that will help user to track and record their finance income and expenses.

Main Problem

Often people don't care much about their personal financial records, so without them realizing it their money just runs out. From this problem I try to solve this problem at least for me personally.

My Role on the Project

Fullstack Developer:

- Design the System Workflow
- Developed the API Backend using Node.js
- Developed the Frontend using React.js
- Designing the Database

Tech Stack

Backend: Node.js

Frontend: React.js

Database: MySQL

Tools: Github, VsCode

II. System Architecture

The HematWeb application is built using a modern, decoupled client-server architecture. This means that the user-facing interface (Frontend) and the data management engine (Backend) are two completely independent applications. They “communicate” with each other through API (Application Programming Interface).

1. Frontend (Client Side)

Technology: React.js

Main Responsibilities: Responsible for everything the user sees and interacts with. This is the “face” of the application. Its tasks include:

- Rendering the user interface (UI) based on data.

- Capturing user input (e.g., filling out the login form or adding a transaction).
- Managing client-side state (e.g., which data is currently displayed, whether a pop-up is open).
- Handling navigation between pages without requiring a full browser refresh (Single Page Application), using React Router.

How It Works: All React code is bundled into static files (HTML, CSS, JavaScript) and runs entirely in the user's browser. The frontend has no direct access to the database for security reasons.

Communication: Uses Axios to make HTTP requests to API endpoints provided by the backend.

2. Backend (Server Side)

Technology: Node.js with the Express.js framework

Main Responsibilities: Acts as the “brain” and intermediary of the system. Its tasks include:

- Receiving requests from the frontend.
- Processing business logic (validating data, performing calculations).
- Handling authentication and authorization to ensure that only legitimate users can access or modify data. This is done by verifying JSON Web Tokens (JWT).
- Communicating directly with the database to retrieve or store data.

How It Works: Runs on a localhost server. It continuously “listens” for incoming requests on a specific address and port.

Communication Bridge: Provides a REST API, a collection of predefined URLs (called endpoints). Examples:

- POST /api/auth/login
- GET /api/transactions

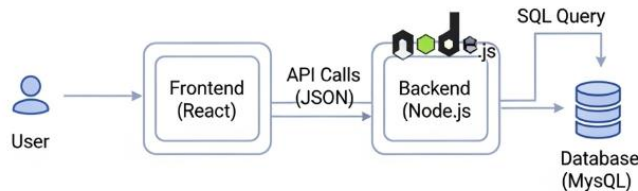
3. Database

Technology: MySQL

Main Responsibilities: Serves as the “storage cabinet” for structured, persistent data. Its sole purpose is to store, retrieve, update, and delete data as instructed by the backend.

How It Works: This database is isolated and only accepts connections from the backend application. This is a critical security layer to protect user data from public access. The frontend never knows where or how this data is stored.

System Architecture Visualisation



III. Key Features

- Secure User Authentication:
 - User registration with validation.
 - Login with credential verification.
 - Passwords are hashed using bcrypt.
 - Session management with JWT and protected API routes.
- Transaction Management (CRUD):
 - Users can create new income or expense records.
 - Users can view a list of all their transactions.
 - Users can Update their Photo Profile.
 - Users can Update their Username.
 - Users can Update their Password.
- Interactive Dashboard:
 - Displays a summary of the current balance.
 - Provides a simple chart of monthly Income and Expense Comparison
 - Provides simple visualizations of total income and expenses.
- Responsive Design:
 - The interface adapts seamlessly to various screen sizes, from desktop to mobile.

IV. Website Visualisation

Login Page

Login HematApp

Login

Don't have an account? [Register here](#)

Register Page

Register HematApp

Register

Already have an account? [Login here](#)

Dashboard Page



Semua
Bulan Ini
Minggu Ini

Total Balance
1 Milyar

Total Income
1,1 Milyar

Total Expense
50,6 Juta

Add New Transaction

Transaction Title

Amount

Expense

Add

Income History

Trf dari Once 350 Juta

Gaji Magang Yogyakarta 520 Juta

Gaji Konser 230 Juta

Expense History


Beli Kabel Jack Gitar X2105 Rp 550.000

Beli Keyboard Roland 50 Juta

Profile Page

Profil Saya

Foto Profil



Choose File | No file chosen

Upload Foto Baru

Ubah Username

Username saat ini: **ahmad123**

ahmad123

Simpan Username

Ubah Password

Password lama

Password baru

Ubah Username

Username saat ini: **ahmad123**

ahmad123

Simpan Username

Ubah Password

Password lama

Password baru

Simpan Password

V. Github

Frontend Source Code: <https://github.com/bensfain/hematweb-frontend>

Backend Source Code: <https://github.com/bensfain/hematweb-backend>