Full Stack Development with MERN

Project Report

Personal Expense Tracker

Team ID: SWTID1743870329

Team Members:

1. Harsh Burman
2. Ayush Pandey
3. Adamya Gupta
4. Yuktika Mishra

1. Introduction

1.1 Project overview

The Personal Expense Tracker App is a digital solution designed to simplify the way users manage their financial activities. It provides a user-friendly interface that allows individuals to effortlessly record daily expenses, categorize them, and analyze their spending habits. With robust functionality, the app supports budget creation, real-time expense tracking, and instant notifications when users approach or exceed their spending limits.

The app features an intuitive design that makes it easy to add new transactions, assign categories, and include relevant details such as payment method, vendor, and transaction date. Users can generate detailed reports and visualize their financial data using graphs and charts for deeper insights into their financial health.

The application also caters to business users by offering support for managing multiple accounts and tracking expenses across various departments or projects. A strong backend infrastructure ensures secure data management, user authentication, and administrative capabilities for monitoring app performance and handling inquiries.

1.2 Purpose

The primary purpose of the **Personal Expense Tracker App** is to empower users to take control of their finances through efficient and organized expense management. The app is designed to promote financial awareness by helping users monitor their daily spending, set realistic budgets, and make informed decisions based on real-time data.

By offering comprehensive tracking and analysis features, the app aims to encourage responsible financial behavior and reduce the risk of overspending. For business users, the app serves as a practical tool for expense tracking and reporting, which aids in budgeting, forecasting, and financial accountability.

2. Ideation phase

2.1 Problem Statement

Managing personal or business expenses efficiently is a growing challenge in today's fast-paced, digitally-driven world. Users often struggle with tracking their daily transactions, controlling their spending, and understanding their financial habits. Existing tools are either too complex, too basic, or lack critical features like real-time tracking, alerts, and actionable insights.

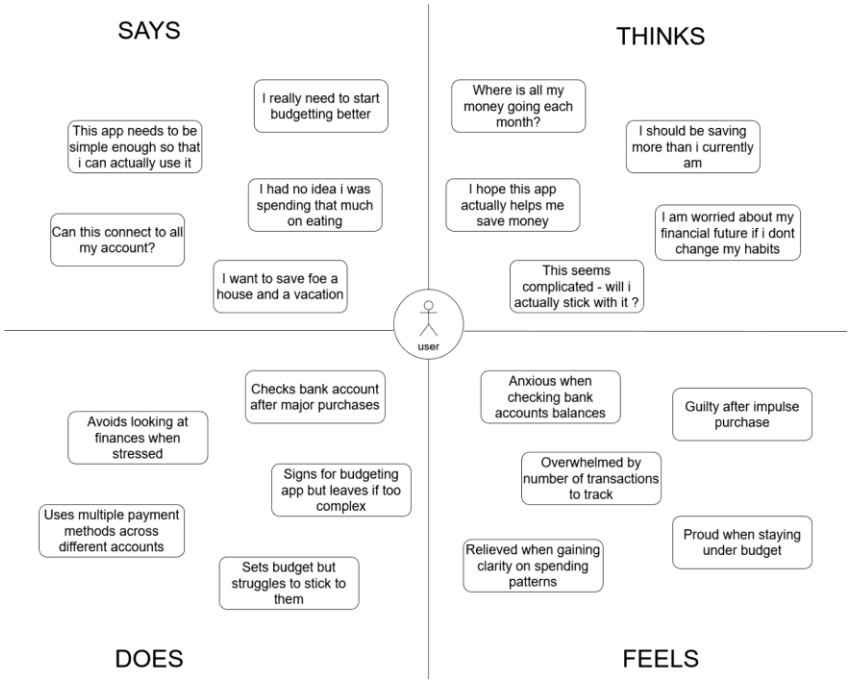
There is a clear need for a secure, user-friendly, and insightful solution that empowers individuals and businesses to manage their finances proactively, make informed decisions, and reduce financial stress.

Problem Statement (PS)	**I am (Customer)**	**I'm trying to**	**But**	**Because**	**Which makes me feel**
PS 1	A working professional managing personal and family expenses	Keep track of my daily expenses and stick to a monthly budget	I often forget to log expenses or lose track of where I'm overspending	Existing tools are either too manual, lack reminders, or are not userfriendly	Frustrated, anxious about money, and out of control of my finances

PS 2	A small business owner managing multiple project expenses	Track spending across different accounts and generate reports for accounting	I don't have a centralized, easy-to-use system to manage and analyze these expenses	Most solutions are costly, overly complex, or not tailored to small businesses	Overwhelmed and inefficient, especially during tax or audit periods
PS 3	A college student trying to manage limited monthly allowance	Monitor my spending on food, travel, and entertainment	I have no idea how much I spend in each category by the end of the month	I don't have a simple tool that helps me categorize or visualize my spending	Lost and often short on money before the month ends
PS4	A tech-savvy user who wants datadriven insights on personal finances	Understand trends in my spending and find areas to save money	My current app doesn't offer visualizations or intelligent analysis	It only records data without offering deeper insights or summaries	Disconnected from my financial goals and less motivated to track

These statements represent a range of user types and their frustrations, helping to clearly define why our Expense Tracker App is needed and what problems it directly solves.

2.2 Empathy Map



An Empathy Map for a personal expense tracker app provides a structured way to capture and visualize user perspectives and needs. It organizes user attitudes and behaviors into four key quadrants: "Thinks," capturing internal thoughts and questions about financial management; "Feels," exploring emotional responses to money tracking and spending habits; "Says," documenting verbal expressions about financial goals and app requirements; and "Does," detailing actual behaviors related to financial management. This visualization tool helps development teams build understanding of user pain points and motivations, allowing them to design features that address true user needs rather than assumptions. By capturing these four dimensions, the empathy map ensures the expense tracker app connects with users' actual financial challenges and aspirations, resulting in a more relevant and engaging user experience.

2.3 Brainstorming

The development of the **Personal Expense Tracker App** began with a detailed brainstorming phase aimed at identifying the key problems users face in managing their finances and exploring practical solutions to address them. This phase involved discussions around user needs, potential features, design approaches, and technology stacks.

Key questions explored during brainstorming:

- What are the common challenges people face in tracking daily expenses?
- How can we make the process of recording and categorizing expenses quick and user-friendly?
- What features would add the most value for both individual and business users?
- How can we present financial data in a clear and meaningful way?
- What security measures are necessary to protect sensitive financial information?
- What technologies and frameworks are most suitable for developing a responsive and scalable app?

Ideas and insights gathered:

- Users want a simple interface with minimal steps to add a transaction.
- Categorization is essential, with support for custom categories and payment methods.
- Setting monthly budgets and receiving alerts when nearing the limit can help control overspending.
- Graphs and visual reports were identified as powerful tools for analyzing spending habits.
- The app should support data export and allow users to generate reports for personal or business use.
- User authentication and data encryption were deemed critical for maintaining data security and trust.

Initial feature list generated:

- Add/edit/delete expenses
- Categorize transactions
- Set and monitor budgets
- View spending trends via charts
- Real-time notifications and alerts

- Secure login and user authentication
- Admin panel for managing users (for business use)

This brainstorming session helped lay a strong foundation for the app's core functionalities and ensured that the final product would align with the real-world needs of its users. It also guided the technical planning and UI/UX design process, ultimately shaping the app into a practical and effective financial management tool.

By focusing on core functionalities first (MVP), we ensured that the app delivers maximum value with minimum complexity. Non-essential but value-adding features were documented for future releases based on user feedback and analytics.

This approach ensured that development remained user-centered, scalable, and aligned with real needs while avoiding feature bloat.

3. Requirement Analysis

3.1 Customer Journey Map

Customer Segments	Scenerios					
Stages	Awareness	Consideration	Onboarding	First use	Regular use	Advocacy
Actions	1. Sees target ad on social media 2. Hears recommendations from friends 3. Searches for Budget help online 4. Reads article about personal finance management	1. Reads app store reviews 2. Watches demo videos 3. Compares features across top options 4. Checks pricing plans	1. Downloads app 2. Creates account 3. Links financial accounts 4. Completes initial setup	1. Explores dashboard 2. Reviews initial spending insights 3. Sets up first budget category 4. Tags recent transactions	1. Reviews weekly spending summaries 2. Categorizes new transactions 3. Adjusts budget categories 4. Checks progress towards goals	1. Shares success stories with friends 2. Leaves positive reviews 3. Recommends app to others 4. Upgrades to premium features
Thoughts/ Feelings	1. Frustrated with current financial situation 2. Hopeful that a solution exists 3. Curious about different options 4. Uncertain about what would work best	1. Concerned about security 2. Evaluating time commitment required 3. Excited about potential benefits 4. Hesitant about subscription cost	1. Anxious about sharing financial data 2. Impatient with lengthy setup 3. Excited to get started 4. Overwhelmed by options	1. Surprised by spending patterns 2. Motivated by easy to understand visuals. 3. Curious about additional features 4. Slightly overwhelmed by data	1. Motivated by progress visualization 2. Developing new financial awareness 3. Building habit of financial tracking	1. Pride in financial progress 2. Gratitude for helpful tips 3. Confidence in financial decisions 4. Trusts in the platfoem
Pain Points	1. Too many options to choose from 2. Difficulty differentiating from similar apps. 3. Unsure if expense tracking will actually help	1. Worry about hidden fees 2. Concern about data sharing prices 3. Uncertainity about ease of use	1. Too many setup steps 2. Confusion about account linking 3. Uncertainity about catigorization settings	1. Incorrect automatic categorization 2. Learning curve for navigation 3. Uncertainity about next step	1. Maintenance feels time consuming. 2. Occasional sync issues with accounts 3. Frustration with budget overruns	1. Limited ways to share achievements 2. Difficulty explaining value to others
Opportunities	1. Clear messaging about unique benefits 2. Aurthentic user testimonials 3. Simple comparision with competitors 4. Free trial or basic version available	1. Transparent pricing 2. Clear security 3. Simple feature comparision chart 4. "How it works" walkthrough	1. Streamlined signup process 2. Clear security reassurance 3. Interactive tutorial 4. Quick start option with defaults	1. Guided first use experience 2. Early wins with immediate insights 3. Simple customization options 4. Proactive help tips	1. Automated customization improvements 2. Streamlined routine tasks 3. Positive reinforcement for good habits 4. Helpful tips based on spending patterns	1. Referral program with benefits 2. Shareable milestone achievements 3. Community features 4. Premium benefits that grow with usage

The Customer Journey Map for a personal expense tracker app chronicles the end-to-end user experience from discovery to long-term engagement. Beginning with Awareness, where potential users first recognize their need for financial management, it progresses through Consideration as they evaluate options and features. The Onboarding stage captures the critical first interactions with the app, while First Use reflects initial explorations and discoveries. Regular Use documents the ongoing relationship users develop with the app as it becomes integrated into their financial routine. Finally, Advocacy represents the stage where satisfied users share their positive experiences. Each stage documents specific actions, thoughts/feelings, pain points, and opportunities, creating a comprehensive blueprint that helps product teams identify moments that matter most to users and design appropriate interventions to enhance satisfaction and retention.

3.2 Solution Requirement

Based on the identified problems, user needs, and insights gathered during brainstorming and user journey mapping, the Personal Expense Tracker App must fulfill a set of well-defined functional and non-functional requirements to provide a valuable and seamless user experience.

1. Functional Requirements

These define the core features and actions the app must perform:

- User Registration & Authentication
  - Secure sign-up/login system
  - Password recovery option
- Expense Management
  - Add, edit, and delete expenses
  - Include details such as date, amount, category, vendor, payment method, and notes
- Categorization & Tagging
  - Predefined categories (e.g., Food, Transport, Bills)
  - Option to add custom categories
- Budget Tracking
  - Set monthly or weekly budgets
  - Visual indicators showing budget consumption
  - Notifications on nearing or exceeding limits
- Reports and Analytics
  - Generate daily, weekly, monthly reports
  - Visual charts to analyze spending trends
  - Download/export reports in PDF or CSV format
- Account Management
  - View and update personal profile
  - Manage multiple accounts (for business use)
- Admin Controls (for Business Users)
  - Admin dashboard to manage user accounts
  - Monitor app usage and data accuracy
  - Respond to user queries

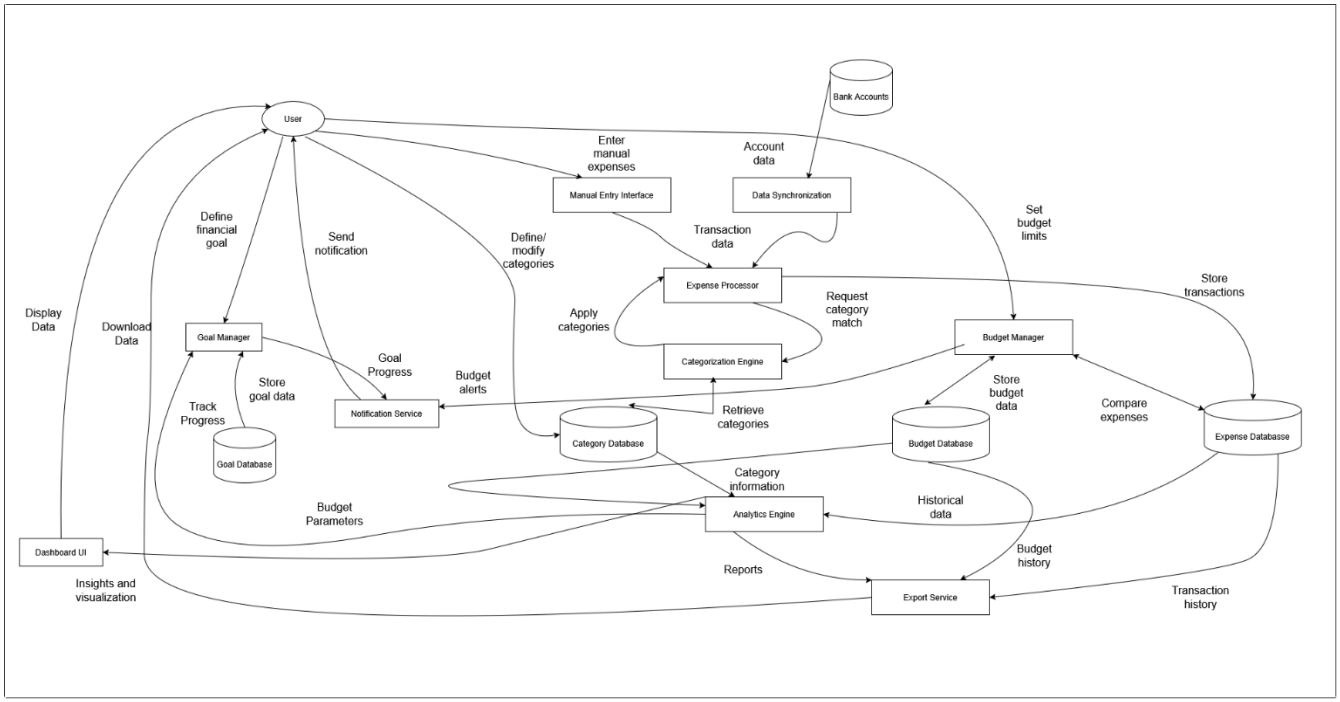
2. Non-Functional Requirements

These define how the system performs and operates:

- Usability
  - Intuitive and responsive UI/UX design

    - Minimal learning curve for new users
- Performance
    - Fast response time for recording transactions and loading data
    - Smooth performance across different devices and screen sizes
- Security
    - Data encryption for sensitive user and financial information
    - Secure authentication methods to prevent unauthorized access
- Scalability
    - Ability to support an increasing number of users and transactions
    - Modular architecture for easy addition of new features
- Reliability
    - System availability with minimal downtime
    - Automatic data backup and recovery mechanisms
- Compatibility
    - Cross-platform support (Android, iOS, and web version)
    - Browser and OS compatibility for web users

3.3 Data Flow Diagram



The Data Flow Diagram (DFD) for a personal expense tracker app visualizes how financial information moves through the system, illustrating both data processing logic and storage requirements. It shows how user inputs and external financial sources like bank accounts feed into core processing components that handle, categorize, and analyze expenses. The diagram maps connections between key data stores (expense, category, budget, and goal databases) and processing elements like the expense processor, categorization engine, analytics engine, and notification service. By tracing data paths from input sources through various transformations to final outputs like dashboards and reports, the DFD helps development teams understand system requirements, identify potential bottlenecks, establish appropriate security measures, and ensure all components communicate effectively. This technical blueprint serves as a foundation for system architecture and database design decisions.

3.4 Technology Stack

Frontend Technologies

ReactJS

- Core library for building dynamic and responsive user interfaces.
- Utilizes React hooks such as useState, useEffect, and useContext for efficient state management and lifecycle handling.

React Router

- Implements client-side routing to navigate between different pages and views.
- Features like useNavigate and ProtectedRoute ensure secure and seamless navigation.

React Icons

- Enhances the UI using icon libraries such as react-icons/lu, react-icons/io, react-icons/bs, and react-icons/wi.

Styling

- Tailwind CSS is likely used as a utility-first CSS framework, based on class names like bg-primary, bg-orange-500, etc.

- Google Fonts integration ensures a customized and consistent typography experience.

Toast Notifications

- react-hot-toast is used to provide user-friendly notifications and alerts across the application.

Custom Components

- Reusable and modular components like InfoCard, Modal, DeleteAlert, and layout components such as DashboardLayout and AuthLayouts help maintain clean and maintainable code architecture.

Backend Communication

Axios

- Handles HTTP requests (GET, POST, DELETE, etc.) to communicate with the backend APIs.

- Facilitates data submission, retrieval, and file downloads such as income/expense reports.

API Paths Configuration

- A centralized API_PATHS configuration ensures efficient management and modification of backend endpoints.

Authentication

Custom Hooks & Context

- Authentication is implemented using custom React hooks like useUserAuth and AuthProvider for managing login, logout, and route protection.

JWT Integration

- JSON Web Tokens (JWT) are used for securing API communications and user-specific route protection.

Utilities

Helper Functions

- Includes utilities such as addThousandsSeparator for number formatting and validateEmail for input validation.

File Upload Utility

- A utility function like uploadImage is used during user signup or profile updates.

Data Management

- React's built-in state management (useState, useContext) is extensively used throughout the app.

- While libraries like Redux or MobX are not used, the state is effectively managed via hooks and context.

Styling Frameworks

Tailwind CSS (assumed)

- Based on utility class usage, Tailwind CSS is likely the primary styling tool for layout, spacing, and color theming.

Development Practices

- Modular component structure and modern development practices are followed.

- Efficient folder hierarchy, centralized configs, and reusable logic contribute to clean code and scalable architecture

Category	**Technology/Library**
Frontend Framework	ReactJS
Routing	React Router
Icons	React Icons(Lucide,Ionicons)
Backend Communication	Axios(Custom Instance)
Authentication	Custom Hook (useUserAuth)
Utilities	Helper Function (addThousandsSeparator)
Styling	Likely Tailwind CSS

4. Project Design

4.1 Problem Solution Fit

The **Personal Expense Tracker App** is designed to directly address the financial management struggles faced by individuals and businesses, particularly in an increasingly cashless and fast-paced world. Through thorough user research, empathy mapping, and journey mapping, a clear behavioral pattern emerged: users often lack real-time control, visibility, and discipline over their daily expenses.

Purpose

To solve complex financial tracking issues in a way that aligns with the real-life behaviors and needs of modern users, ensuring our solution is accessible, relevant, and easy to adopt.

Problem–Solution Fit Analysis

Problem	**Solution**
Users often forget to track daily expenses due to time-consuming processes	Easy-to-use, intuitive interface to quickly log transactions on the go
Lack of insights into spending patterns	Visual reports and charts to help users understand where their money is going
Overspending due to poor budget control	Budget setting feature with real-time notifications when users approach or exceed limits
Manual tracking tools are inefficient and prone to error	Automated data storage, retrieval, and categorization through a cloud-based backend
Businesses need multi-account expense tracking and reporting	Admin features, multiple user support, and downloadable reports for different

	projects or teams
Users fear sharing financial data online due to privacy concerns	End-to-end encryption, secure login (JWT), and encrypted password storage (bcrypt)

Key Fit Indicators

- Behavioral Insight: Users already try to manage expenses manually or via basic apps — we're improving and simplifying this existing behavior.

- Frequent Annoyance: Forgetting to track expenses, not knowing where money went — solved via quick inputs and categorized history.

- Urgent/Costly Problem: Overspending leads to stress and financial instability — solved by budget tracking and real-time alerts.

- Trust Factors: Secure logins, encrypted storage, and personal data privacy — key to user adoption and long-term usage.

- Multi-channel Access: Mobile-friendly UI ensures access anytime, anywhere — aligned with current mobile-first behavior.

Outcomes and Benefits

- Succeed Faster: Rapid user onboarding due to familiar interaction flows and intuitive design.

- Increased Adoption: By aligning with user behavior (mobile usage, desire for simplicity, real-time updates), the app becomes part of their daily routine.

- Better Communication Strategy: Messaging focuses on solving frequent money leaks and stress, using empathetic and solution-oriented language.

- Improved Trust and Loyalty: Transparency, security, and consistent problem-solving lead to long-term engagement.

4.2 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	● Many individuals and businesses struggle with tracking daily expenses and maintaining financial discipline. ● Traditional methods like spreadsheets or manual logs are inefficient, error-prone, and lack real-time insights. ● Users lack an integrated, secure, and user-friendly platform that provides analytics, alerts, and comprehensive budget management.
2.	Idea / Solution description	● Develop a full-stack **Personal Expense Tracker App** using the **MERN** stack.

		• Provide features such as expense logging, category-wise tracking, budgeting, real-time notifications, and detailed analytics through charts. • Secure login system with encryption, allowing users to manage multiple accounts (for personal or business use). • Admin panel for handling user queries, managing data accuracy, and performance monitoring.
3.	Novelty / Uniqueness	• Combines real-time budget alerts, visual spending insights, and business-level expense management into one platform. • Offers a clean, responsive UI with custom categories, downloadable reports, and personalized dashboards. • Scalable architecture using MERN stack that supports both personal and enterprise use cases. • Emphasis on data security with features like password hashing and JWT-based session management.

4.	Social Impact / Customer Satisfaction	• Promotes financial literacy and responsible spending habits among users. • Helps individuals avoid debt and financial stress by providing clarity on where their money is going. • For businesses, simplifies expense reporting, tracking departmental budgets, and improving accountability. • Offers a seamless user experience and builds customer trust through transparency and security.
5.	Business Model (Revenue Model)	**Freemium Model:** • Free access to basic features. • Subscription plans for premium features like advanced analytics, multi-account management, data export, and custom categories. **In-App Ads (Optional):**

		• Non-intrusive ads for free users. **Enterprise Licensing:** • Offer tailored plans for businesses to manage multiple employees or projects. **Affiliate Marketing:** 6. Partner with financial services like budgeting tools, investment apps, or insurance providers.
7.	Scalability of the Solution	**Technical Scalability:** • Built on a modular MERN architecture to support thousands of users with smooth performance. • MongoDB Atlas supports scalable cloud-based database solutions. **Feature Scalability:** • Easily extensible to include AI-based budget suggestions,

		goal setting, or bill reminders. **Market Scalability:** ● Can target individual users, families, and businesses. ● Potential to expand into related domains like savings planning, credit monitoring, or investment tracking.

4.3 Solution Architecture

The solution architecture consists of multiple layers working together to provide a seamless, secure, and scalable user experience. The app follows a modular, RESTful, and component-based architecture with a clear separation of concerns.

Client-Side (Frontend) – React.js

Responsibilities:

○ Provides a responsive and interactive user interface

○ Handles routing with React Router

○ Sends API requests to the backend via Axios/Fetch

○ Displays visual insights using libraries like Chart.js or Recharts

Components:

○ Dashboard (summary view)

- Expense form (add/edit)
- Expense list (filtered by date/category)
- Budget manager
- Report generator
- Authentication (login/register forms)

State Management:

- React Hooks (useState, useEffect)
- Optionally, Redux or Context API for global state (e.g., user session)

Server-Side (Backend) – Node.js + Express.js

Responsibilities:

- Handles all business logic
- Processes API requests and routes them accordingly
- Validates input data
- Authenticates and authorizes users (JWT)
- Interfaces with MongoDB for CRUD operations

API Endpoints:

- /api/auth/register – User registration
- /api/auth/login – User login

- /api/expenses – Add/view/edit/delete expenses
- /api/budgets – Set and retrieve budget limits
- /api/reports – Generate summaries and insights
- /api/users – Profile management

Database – MongoDB (with Mongoose)

Responsibilities:

- Stores all persistent data (users, expenses, budgets, etc.)
- Offers schema flexibility for evolving needs

Collections:

- users: userId, name, email, password (hashed), settings
- expenses: expenseId, userId, category, amount, date, vendor, payment method
- budgets: budgetId, userId, category, limit, time range
- reports: generated summaries or saved snapshots (optional)

Authentication & Security

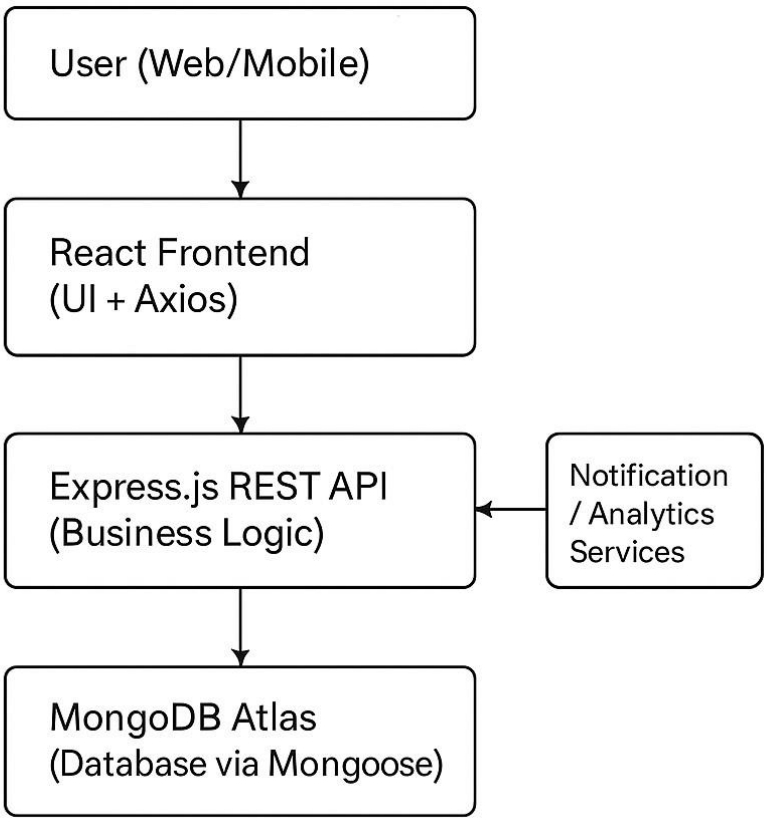
- JWT (JSON Web Tokens): Used for stateless user authentication

- bcrypt.js: Password hashing and storage
- Helmet.js: Sets secure HTTP headers
- CORS Middleware: Controls cross-origin access
- Role-Based Access Control: Optional – Admin/Business user handling

Deployment Architecture

- Frontend Deployment:
  - Vercel / Netlify
- Backend Deployment:
  - Render / Railway / Heroku / AWS EC2
- Database Hosting:
  - MongoDB Atlas (Cloud-based)
- CI/CD (optional):
  - GitHub Actions or GitLab CI for continuous integration & deployment

Logical flow :



5. Project Planning and Scheduling

5.1 Project Planning

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Point	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	A
Sprint-1	Registration	USN-2	As a user, I will receive confirmation email once I have registered for the application.	1	High	A
Sprint-2	Registration	USN-3	As a user, I can register for the application through	2	Low	B

			Facebook.			
Sprint-1	Registration	USN-4	As a user, I can register for the application through Gmail.	2	Medium	A
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	1	High	A
Sprint-2	Dashboard	USN-6	As a user, I can view an overview of my current month's income, expenses, and budget tracking.	3	High	B
Sprint-2	Expense Management	USN-7	As a user, I can add a new expense with details like category, date, amount, and notes.	3	High	B
Sprint-3	Expense Management	USN-8	As a user, I can edit or delete an existing expense entry.	2	High	C

Sprint-3	Reports	USN-9	As a user, I can generate monthly reports and visualize data using graphs.	3	Medium	D
Sprint-4	Budget	USN-10	As a user, I	4	High	D
	Management		can set a budget for categories and get notified when approaching limits.			

Project Tracker, Velocity & Burndown Chart:

Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date (Actual)**
Sprint-1	6	6 days	17 March 2025	22 March 2025	6	22 March 2025
Sprint-2	6	6 days	24 March 2025	29 March 2025	6	29 March 2025
Sprint-3	6	6 days	3 April 2025	8 April 2025	6	8 April 2025
Sprint-4	6	6 days	9 April 2025	14 April 2025	6	14 April 2025

Velocity:

Sprint Duration: 6 Days

Average Velocity (AV) = Total Story Points / Sprint Duration

Example (Sprint-1):
AV = 6 / 6 = 1 story point per day
Overall average velocity can be calculated from completed sprints: Total Story Points (completed): 6 + 6 + 5 = 17

Total Days: 6 * 3 = 18

Overall AV = 17 / 18 ≈ 0.94 story points/day

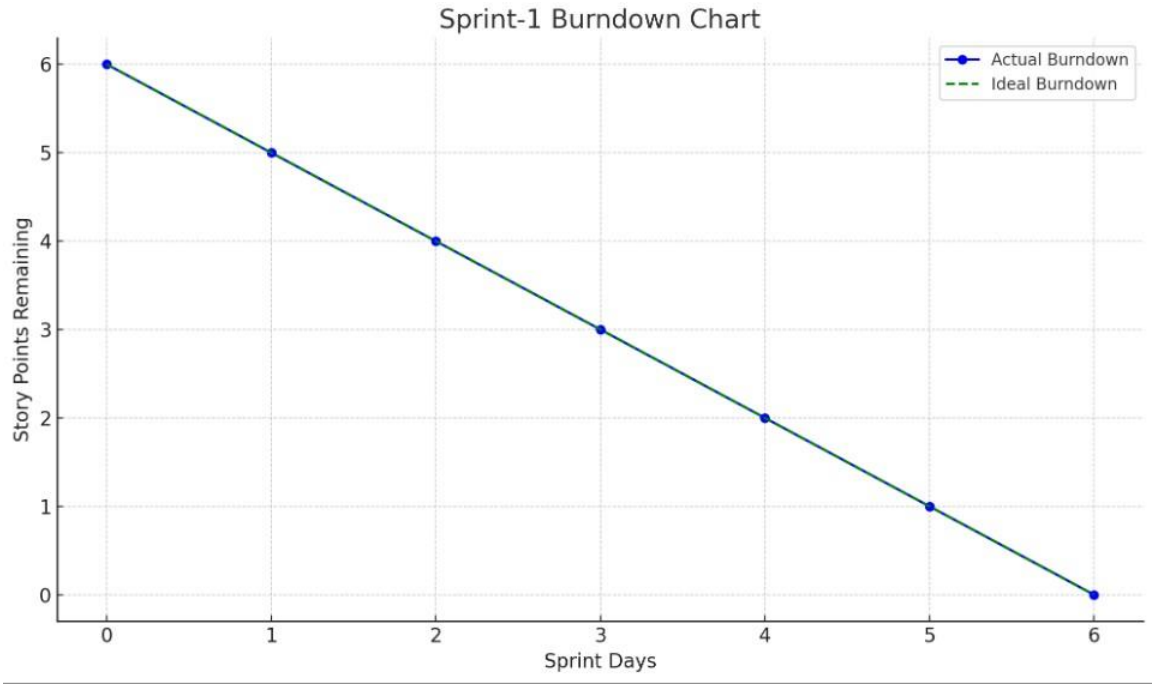
Burndown Chart Overview :

The Burndown Chart plots:

- X-Axis: Sprint Days (Day 1 to Day 6)
- Y-Axis: Story Points Remaining

For each sprint:

- Day 0: Full Story Points
- Each day reduces based on tasks completed



Textual representation for Sprint-1:

Day	**Story points remaining**
Day 0	6
Day 1	5
Day 2	4
Day 3	3
Day 4	2
Day 5	1
Day 6	0

6. Functional and Performance Testing

6.1 Performance Testing

User Acceptance Testing (UAT):

Project Overview:

- Project Name: Personal Expense Tracker App
- Project Description: A full-stack MERN-based web application designed to help users record, categorize, and analyze their personal or business expenses, manage budgets, and gain financial insights through visual reports.
- Project Version: 1.0.0
- Testing Period: April 10, 2025 – April 15, 2025

Testing Scope:

Features and Functionalities to be Tested:

- User registration and login
- Adding, editing, and deleting expenses
- Expense categorization
- Budget setup and notifications
- Report generation and chart visualization
- Data security and session management

User Stories / Requirements to be Tested:

- As a user, I should be able to create an account and securely log in.
- As a user, I want to add daily expenses and categorize them.
- As a user, I want to set budget limits and get alerts when exceeded.
- As a user, I want to view reports with graphical insights.
- As a user, I want my data to be securely stored and accessible only to me.

Test Cases:

Test Case ID	**Test Scenario**	**Test Steps**	**Expected Result**	**Actual Result**	**Actual Result**
TC-001	User Registration	Navigate to Register → Enter details → Submit form	Account should be created and redirected to dashboard	Account created successfully	Pass
TC-002	User Login	Go to Login → Enter valid credentials → Click Login	User should be logged in and directed to dashboard	Successful login	Pass
TC-003	Add Expense	Click "Add Expense" → Fill form → Submit	Expense appears in list under correct category	Expense listed correctly	Pass

TC-004	Edit Expense	Click edit icon →	Updated expense	Expense updated	Pass
		Modify details → Save changes	should reflect correctly	successfully	
TC-005	Delete Expense	Click delete icon → Confirm deletion	Expense should be removed from the list	Expense removed	Pass
TC-006	Budget Setup and Alert	Go to Budgets → Set category limit → Add expenses to exceed budget	Alert should trigger when limit is crossed	Expense removed	Pass
TC-007	Generate Report	Navigate to Reports → Select date range → Generate	Chart displays accurate visual of categorized expenses	Graph displayed correctly	Pass
TC-008	Unauthorize d Access	Access dashboard URL without logging in	Redirect to login page	Redirected to login	Pass
TC-009	Logout Functionality	Click logout button	User should be logged out and redirected to login page	Logout successful	Pass

Bug Tracking:

Bug ID	Bug Description	Steps to Reproduce	Severity	Status	Additional Feedback
BG-001	Chart not displaying	Reports → Select no	Low	Closed	Added validation to
	for empty date range	data range → Generate Report			avoid empty graphs
BG-001	Budget alert not appearing in real-time	Add expenses quickly above budget limit without refresh	Medium	In Progress	Implementing dynamic budget check
BG-001	Category dropdown not scrollable on mobile	Add Expense → Category → Try scrolling list on mobile device	Medium	Open	Needs better responsive design for mobile

Sign-off:

Tester Name: Aditya Gautam

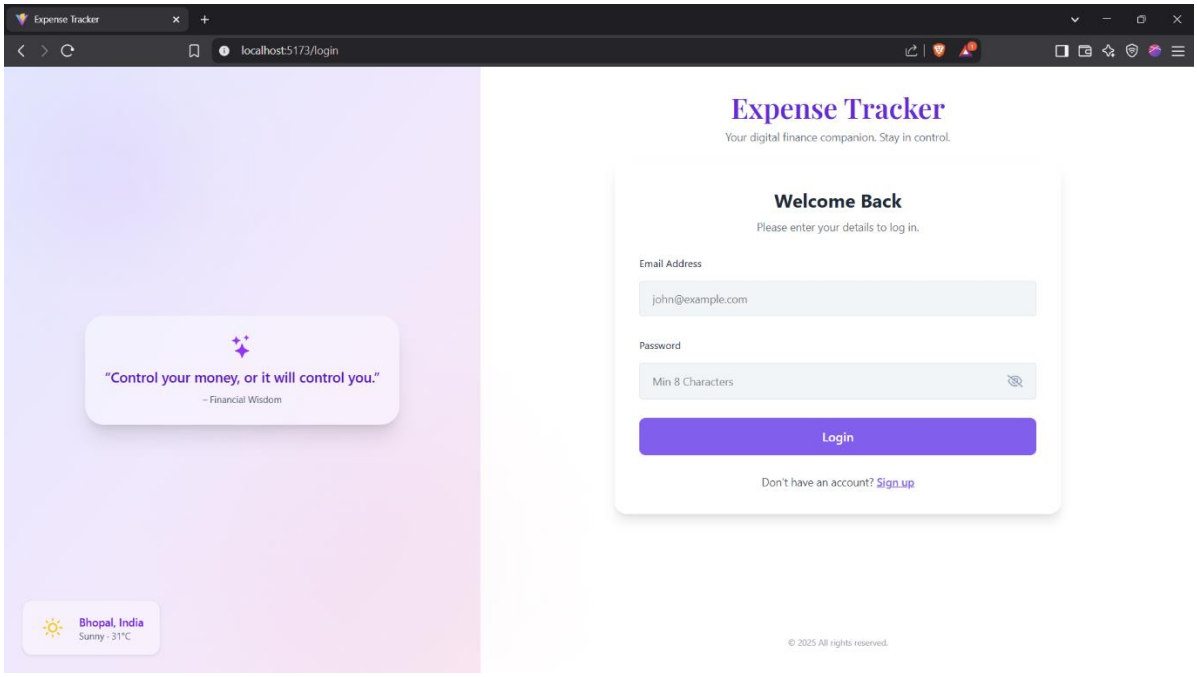
Date: April 13, 2025

Signature:

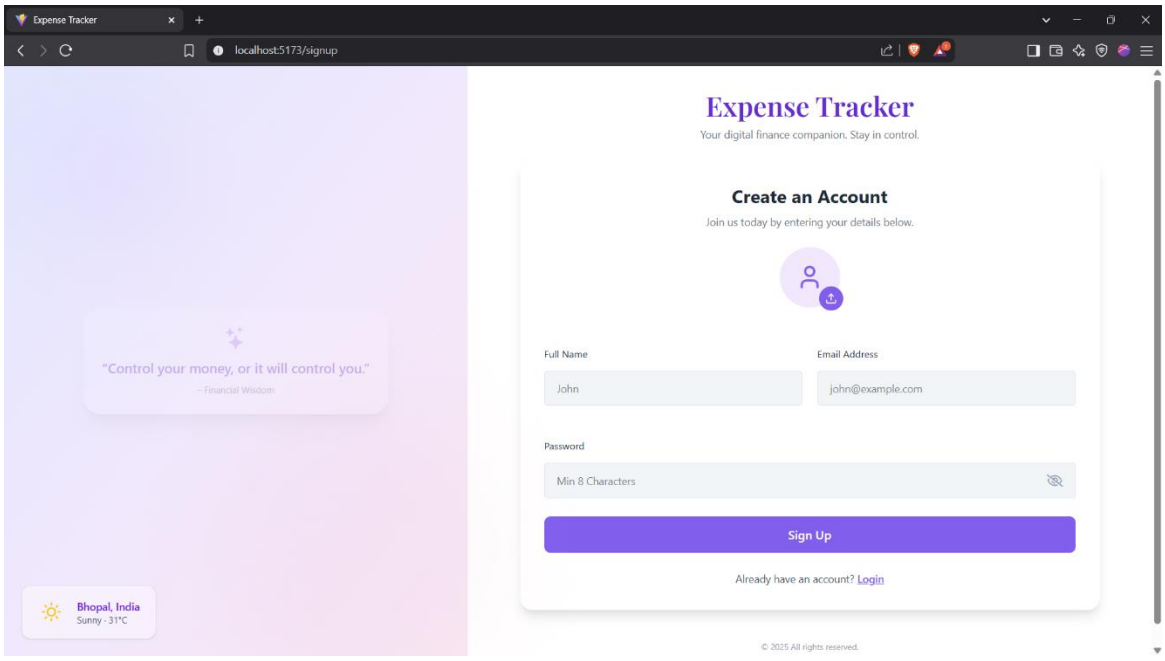
7. Results

7.1 Output Screenshots

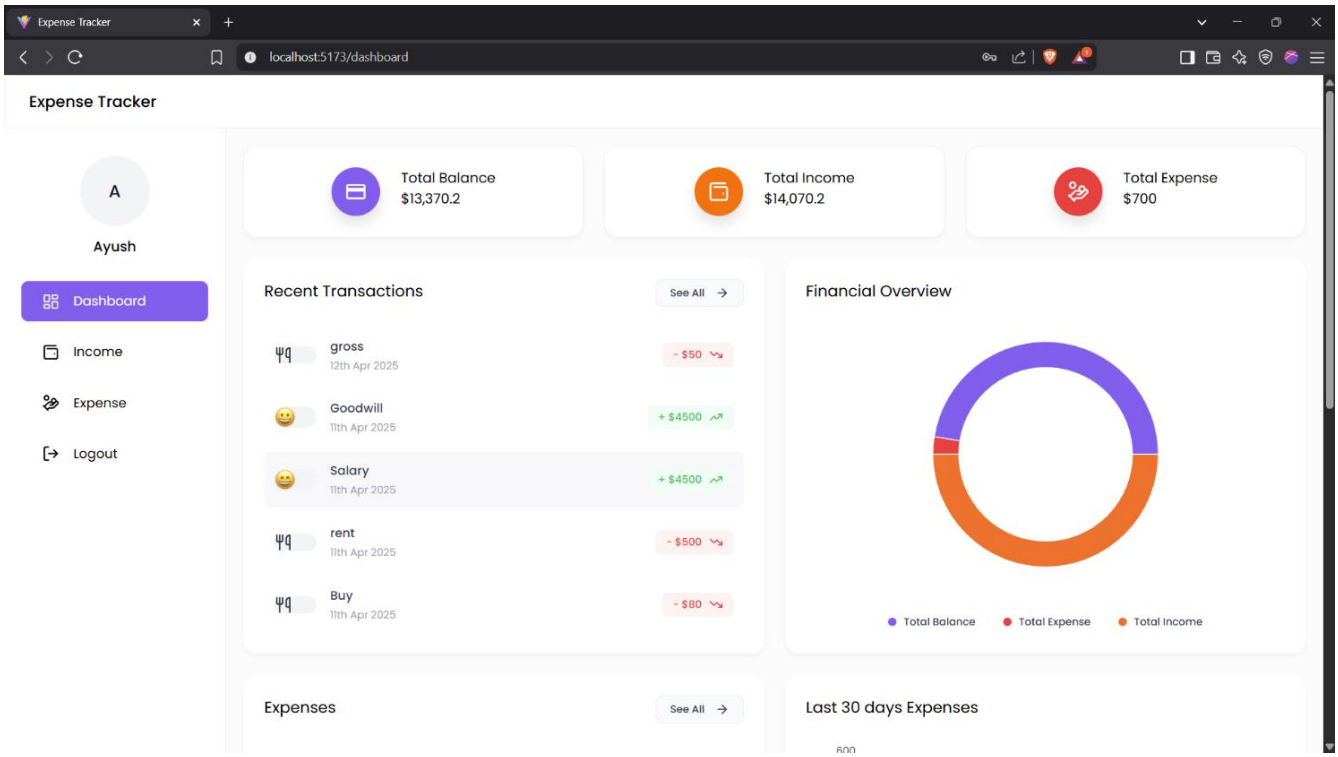
Login :

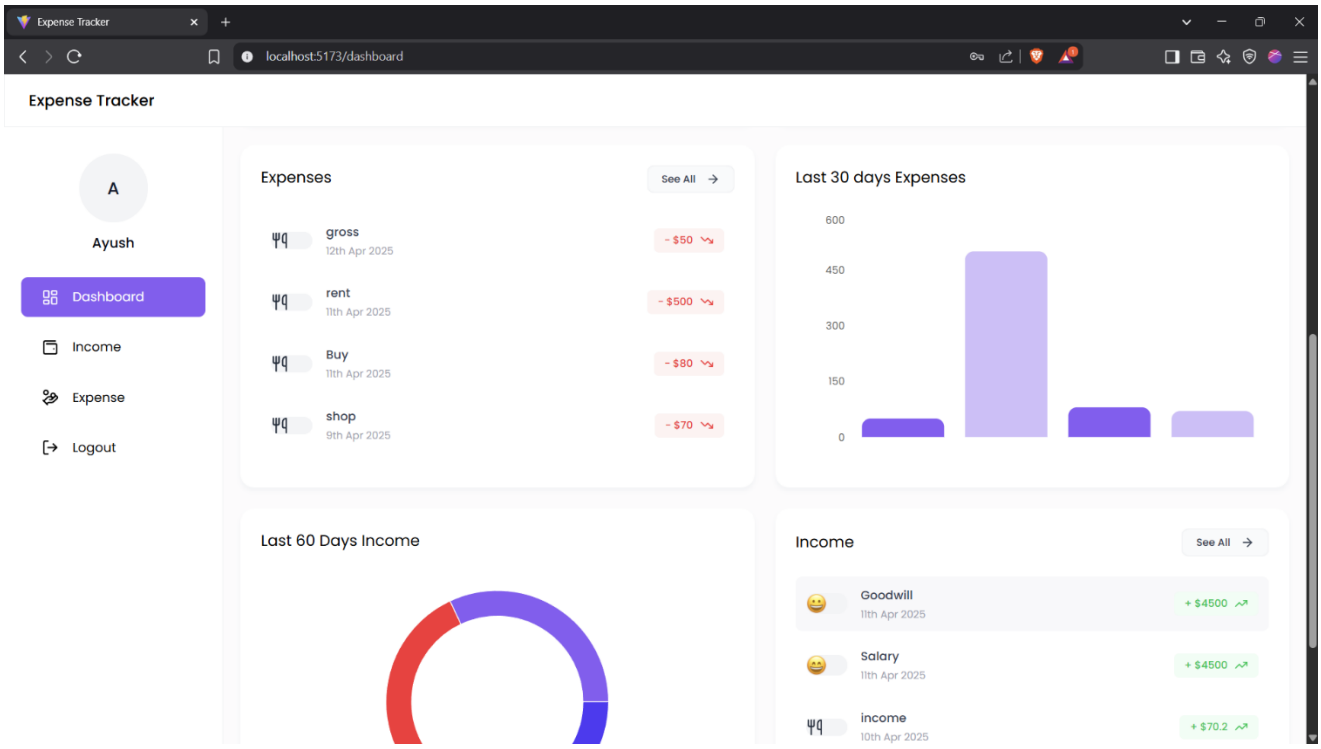


Signup :

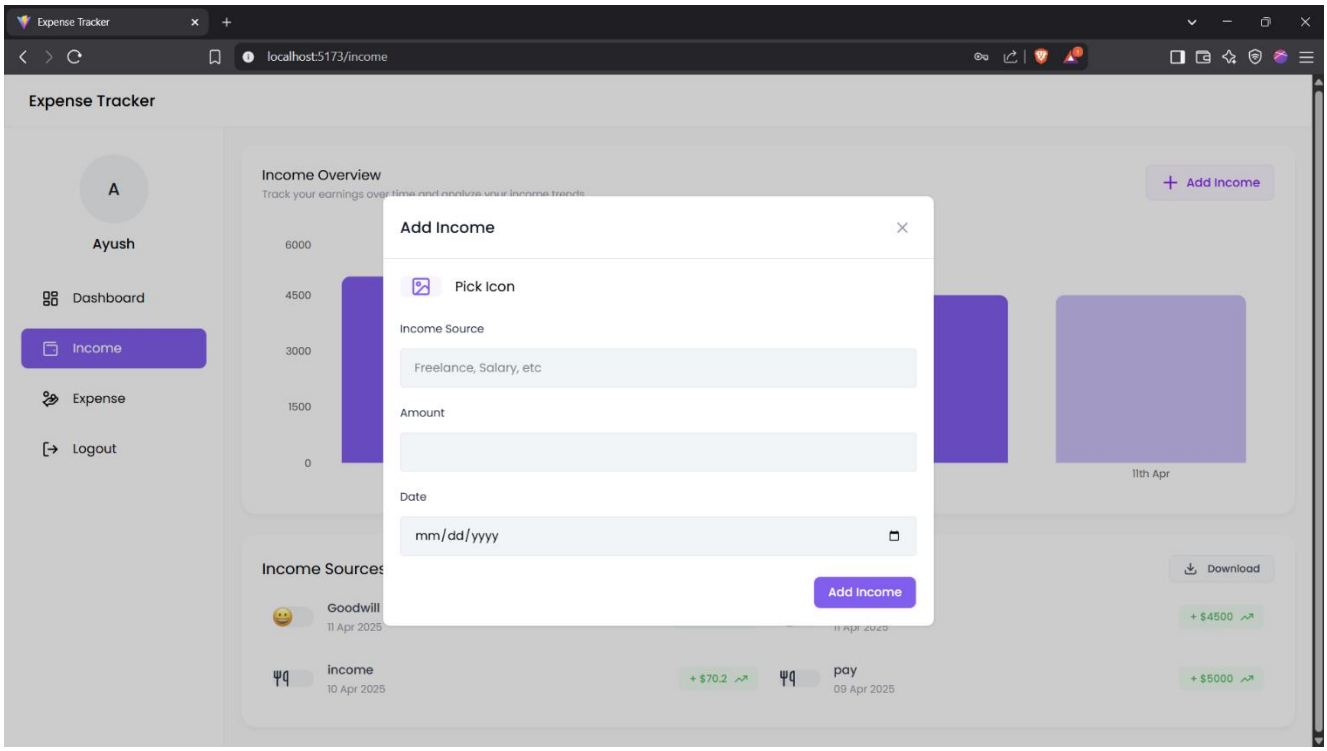


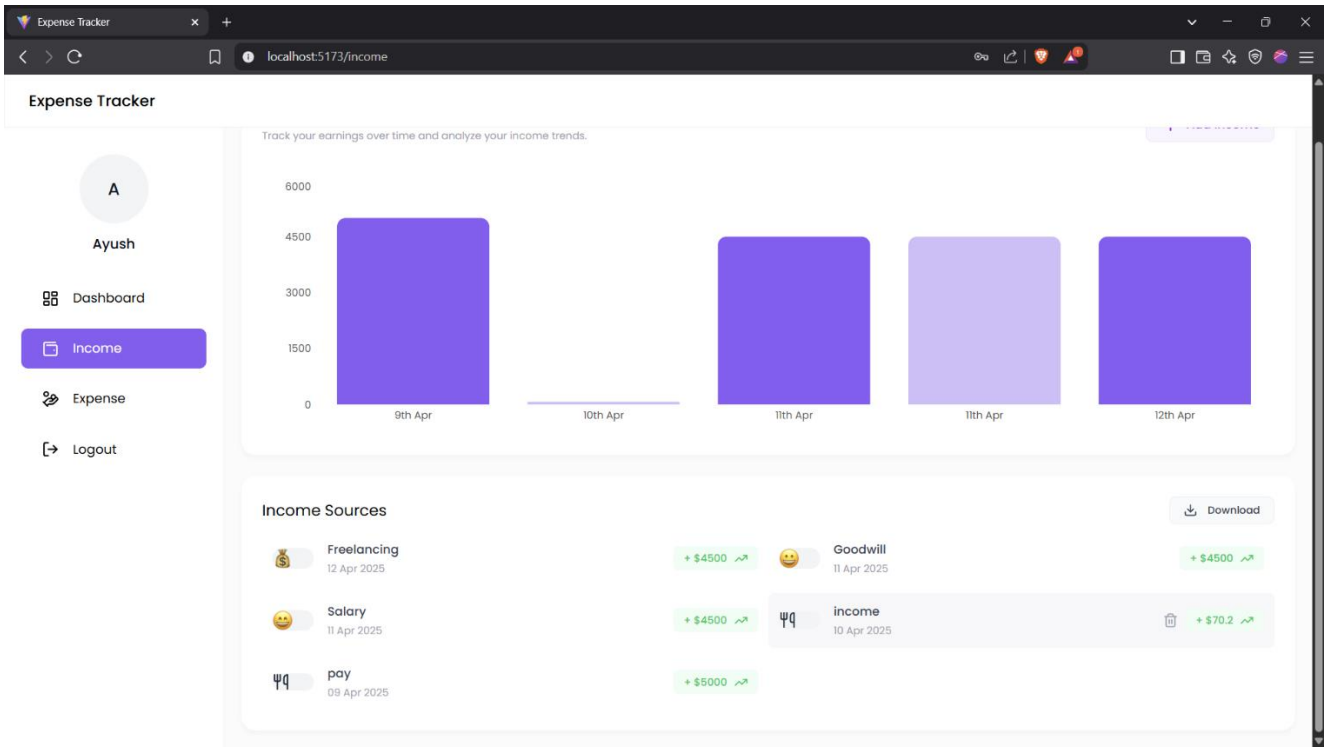
Dashboard:

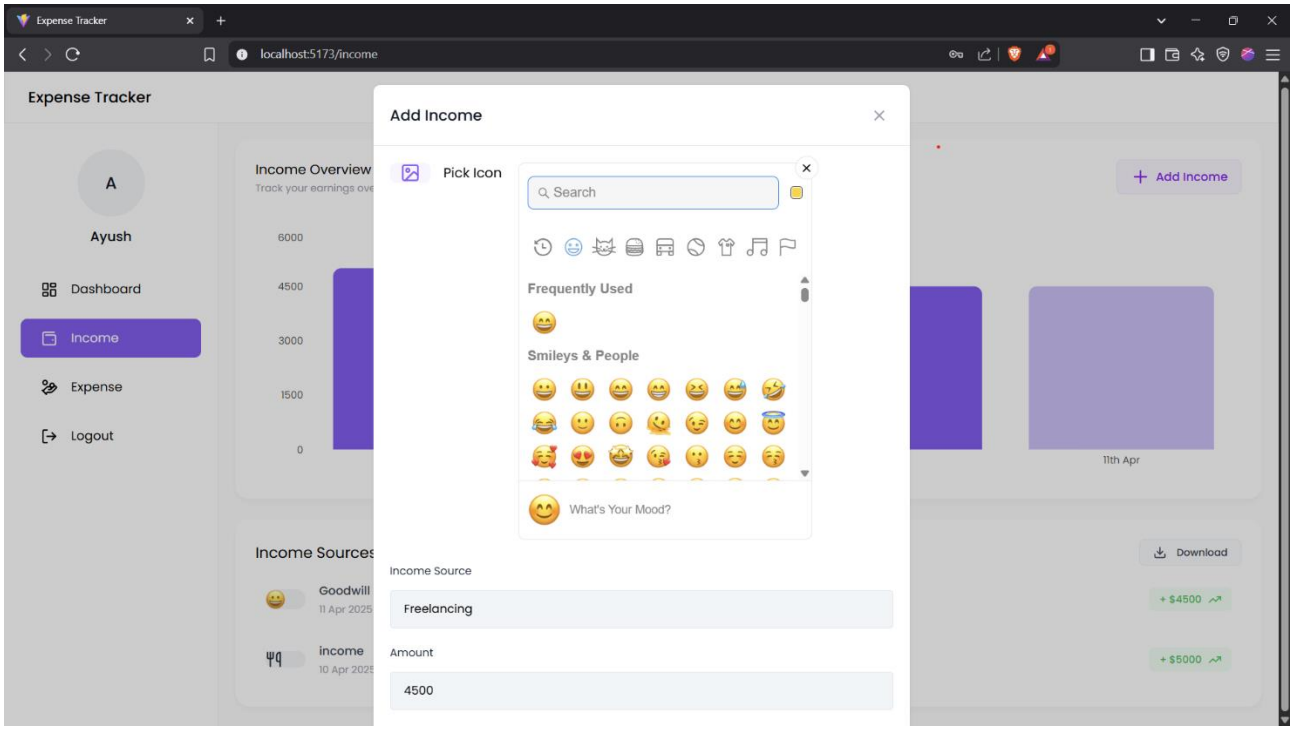

Expense Tracker
localhost:5173/dashboard
Expense Tracker
A
Ayush
Dashboard
Income
Expense
Logout
Total Balance
$13,370.2
Total Income
$14,070.2
Total Expense
$700
Recent Transactions
See All
gross
12th Apr 2025
- $50
Goodwill
11th Apr 2025
+ $4500
Salary
11th Apr 2025
+ $4500
rent
11th Apr 2025
- $500
Buy
11th Apr 2025
- $80
Financial Overview
Total Balance
Total Expense
Total Income
Expenses
See All
Last 30 days Expenses
600

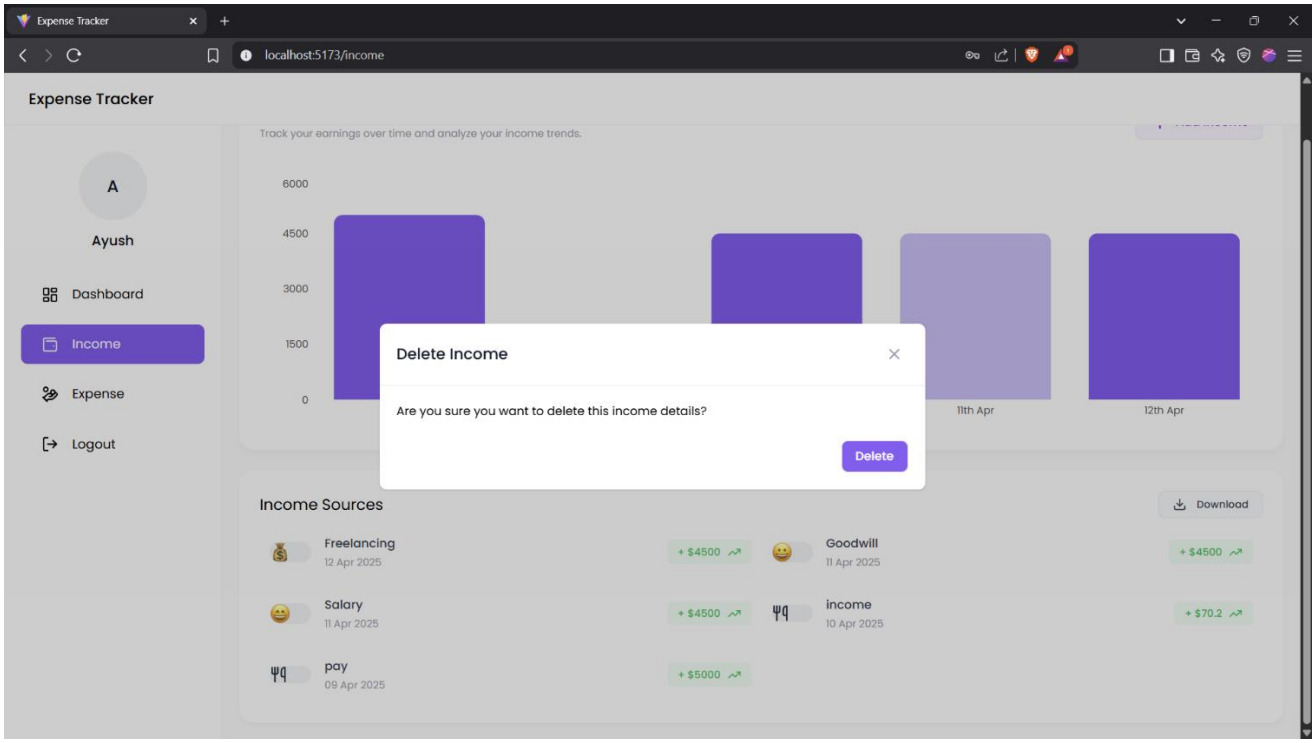

Expense Tracker
localhost:5173/dashboard
Expense Tracker
A
Ayush
Dashboard
Income
Expense
Logout
Expenses
See All
gross
12th Apr 2025
- $50
rent
11th Apr 2025
- $500
Buy
11th Apr 2025
- $80
shop
9th Apr 2025
- $70
Last 30 days Expenses
600
450
300
150
0
Last 60 Days Income
Income
See All
Goodwill
11th Apr 2025
+ $4500
Salary
11th Apr 2025
+ $4500
income
10th Apr 2025
+ $70.2

Income :

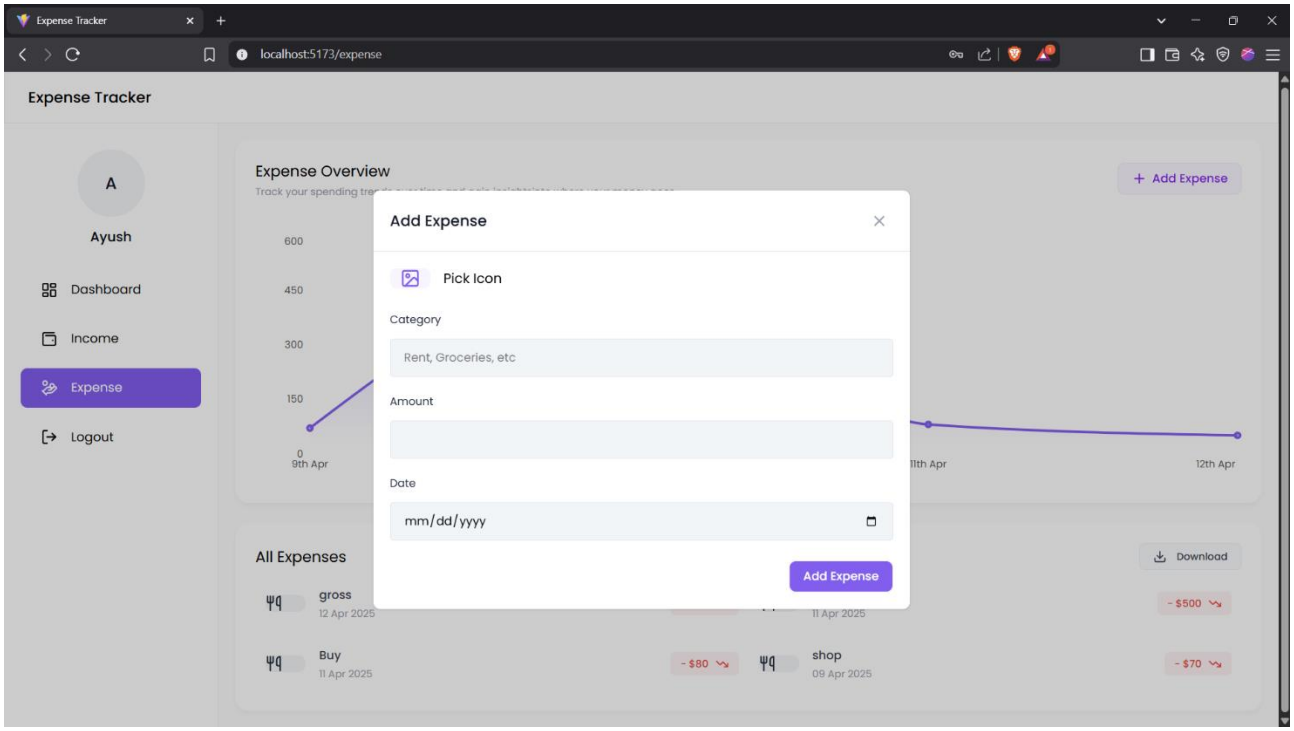

Expense Tracker
localhost:5173/income
Expense Tracker
A
Ayush
Dashboard
Income
Expense
Logout
Income Overview
Track your earnings over time and analyze your income trends.
Add Income
Add Income
Pick Icon
Income Source
Freelance, Salary, etc
Amount
Date
mm/dd/yyyy
Add Income
6000
4500
3000
1500
0
11th Apr
Income Sources
Download
Goodwill
11 Apr 2025
+ $4500
income
10 Apr 2025
+ $70.2
pay
09 Apr 2025
+ $5000

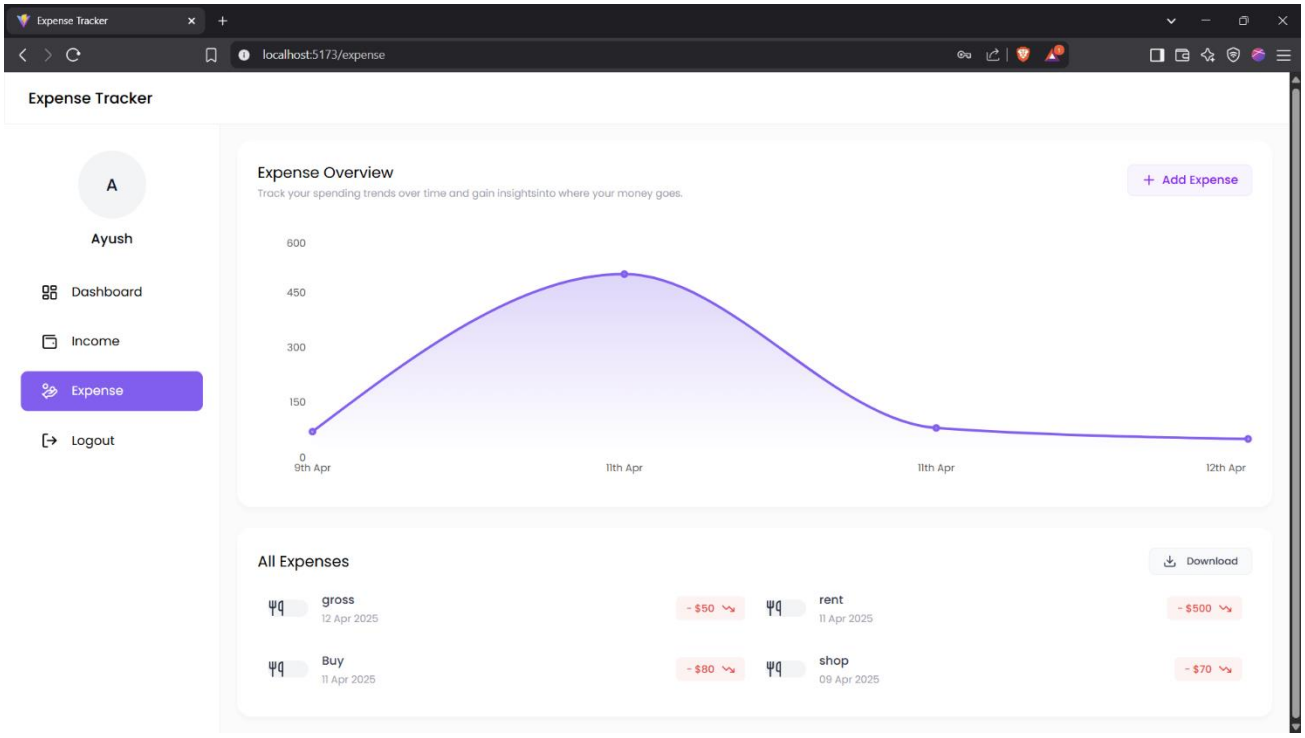

Expense Tracker
localhost:5173/income
Expense Tracker
Track your earnings over time and analyze your income trends.
A
Ayush
Dashboard
Income
Expense
Logout
6000
4500
3000
1500
0
9th Apr
10th Apr
11th Apr
11th Apr
12th Apr
Income Sources
Download
Freelancing
12 Apr 2025
+ $4500
Goodwill
11 Apr 2025
+ $4500
Salary
11 Apr 2025
+ $4500
income
10 Apr 2025
+ $70.2
pay
09 Apr 2025
+ $5000


Expense Tracker
localhost:5173/income
Expense Tracker
Add Income
Pick Icon
Search
Frequently Used
Smileys & People
What's Your Mood?
Income Source
Freelancing
Amount
4500
Income Overview
Add Income
A
Ayush
Dashboard
Income
Expense
Logout
6000
4500
3000
1500
0
11th Apr
Income Sources
Download
Goodwill
11 Apr 2025
+ $4500
income
10 Apr 2025
+ $5000

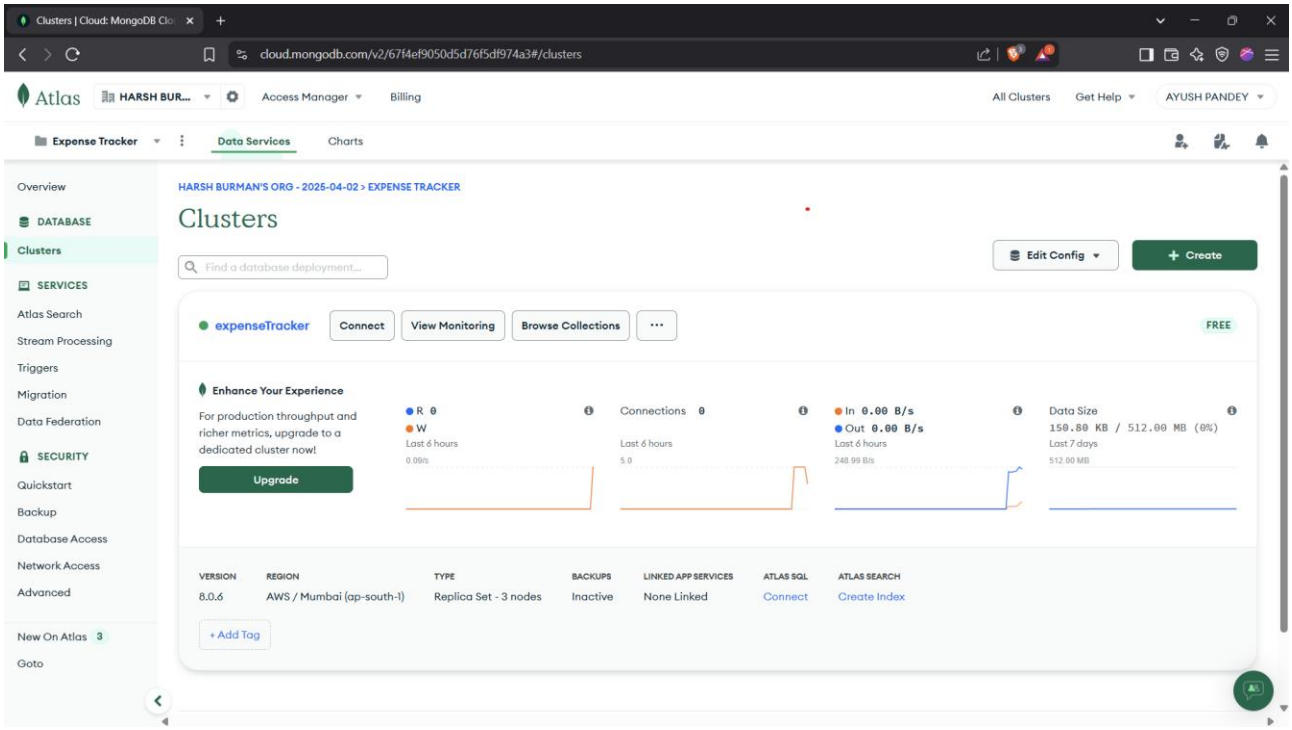

Expense Tracker
localhost:5173/income
Expense Tracker
Track your earnings over time and analyze your income trends.
A
Ayush
Dashboard
Income
Expense
Logout
6000
4500
3000
1500
0
11th Apr
12th Apr
Delete Income
Are you sure you want to delete this income details?
Delete
Income Sources
Download
Freelancing
12 Apr 2025
+ $4500
Goodwill
11 Apr 2025
+ $4500
Salary
11 Apr 2025
+ $4500
income
10 Apr 2025
+ $70.2
pay
09 Apr 2025
+ $5000

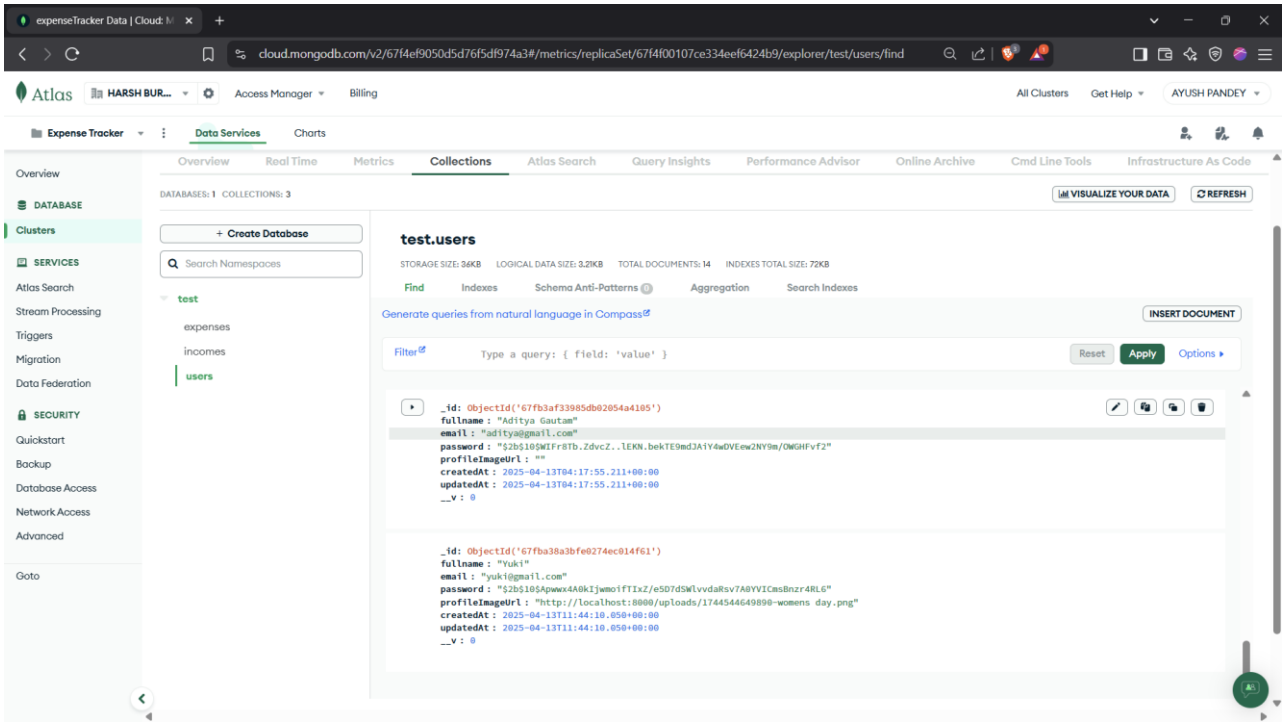
Expense:


Expense Tracker
localhost:5173/expense
Expense Tracker
A
Ayush
Dashboard
Income
Expense
Logout
Expense Overview
Add Expense
Add Expense
Pick Icon
Category
Rent, Groceries, etc
Amount
Date
mm/dd/yyyy
Add Expense
All Expenses
Download
gross
12 Apr 2025
Buy
11 Apr 2025
- $80
shop
09 Apr 2025
- $70
- $500
600
450
300
150
0
9th Apr
11th Apr
12th Apr


Expense Tracker
localhost:5173/expense
Expense Tracker
A
Ayush
Dashboard
Income
Expense
Logout
Expense Overview
Track your spending trends over time and gain insightsinto where your money goes.
Add Expense
600
450
300
150
0
9th Apr
11th Apr
11th Apr
12th Apr
All Expenses
Download
gross
12 Apr 2025
- $50
rent
11 Apr 2025
- $500
Buy
11 Apr 2025
- $80
shop
09 Apr 2025
- $70

Database:


Clusters | Cloud: MongoDB Clo
cloud.mongodb.com/v2/67f4ef9050d5d76f5df974a3#/clusters
Atlas
HARSH BUR...
Access Manager
Billing
All Clusters
Get Help
AYUSH PANDEY
Expense Tracker
Data Services
Charts
Overview
DATABASE
Clusters
SERVICES
Atlas Search
Stream Processing
Triggers
Migration
Data Federation
SECURITY
Quickstart
Backup
Database Access
Network Access
Advanced
New On Atlas 3
Goto
HARSH BURMAN'S ORG - 2025-04-02 > EXPENSE TRACKER
Clusters
Find a database deployment...
Edit Config
+ Create
expenseTracker
Connect
View Monitoring
Browse Collections
FREE
Enhance Your Experience
For production throughput and richer metrics, upgrade to a dedicated cluster now!
Upgrade
R 0
W
Last 6 hours
Connections 0
Last 6 hours
In 0.00 B/s
Out 0.00 B/s
Last 6 hours
Data Size
150.80 KB / 512.00 MB (0%)
Last 7 days
VERSION
REGION
TYPE
BACKUPS
LINKED APP SERVICES
ATLAS SQL
ATLAS SEARCH
8.0.6
AWS / Mumbai (ap-south-1)
Replica Set - 3 nodes
Inactive
None Linked
Connect
Create Index
+ Add Tag


expenseTracker Data | Cloud:
cloud.mongodb.com/v2/67f4ef9050d5d76f5df974a3#/metrics/replicaSet/67f4f00107ce334eef6424b9/explorer/test/users/find
Atlas
HARSH BUR...
Access Manager
Billing
All Clusters
Get Help
AYUSH PANDEY
Expense Tracker
Data Services
Charts
Overview
Real Time
Metrics
Collections
Atlas Search
Query Insights
Performance Advisor
Online Archive
Cmd Line Tools
Infrastructure As Code
DATABASES: 1 COLLECTIONS: 3
VISUALIZE YOUR DATA
REFRESH
+ Create Database
Search Namespaces
test
expenses
incomes
users
test.users
STORAGE SIZE: 36KB
LOGICAL DATA SIZE: 3.21KB
TOTAL DOCUMENTS: 14
INDEXES TOTAL SIZE: 72KB
Find
Indexes
Schema Anti-Patterns
Aggregation
Search Indexes
Generate queries from natural language in Compass
INSERT DOCUMENT
Filter
Type a query: { field: 'value' }
Reset
Apply
Options
_id: ObjectId('67fb3af33985db02054a4105')
fullname : "Aditya Gautam"
email : "aditya@gmail.com"
password : "$2b$10$WIFr8Tb.ZdvcZ..lEKN.bekTE9md3AiY4wDVEew2NY9m/OWGHFvf2"
profileImageUrl : ""
createdAt : 2025-04-13T04:17:55.211+00:00
updatedAt : 2025-04-13T04:17:55.211+00:00
__v : 0
_id: ObjectId('67fba38a3bfe0274ec014f61')
fullname : "Yuki"
email : "yuki@gmail.com"
password : "$2b$10$Apwwx4A0kIjwmoifTIxZ/e5D7dSWlvvdaRsv7A0YVICmsBnzr4RL6"
profileImageUrl : "http://localhost:8000/uploads/1744544649890-womens day.png"
createdAt : 2025-04-13T11:44:10.050+00:00
updatedAt : 2025-04-13T11:44:10.050+00:00
__v : 0

8. Advantages and Disadvantages

Advantages

1. User-Friendly Interface:
    - Intuitive and responsive design ensures users can easily navigate and manage their expenses.
2. Real-Time Notifications:
    - Users receive alerts when they approach or exceed their budget, helping to prevent overspending.
3. Secure Authentication:
    - Strong authentication mechanisms (e.g., JWT, encrypted passwords) protect user data and privacy.
4. Customizable Expense Categories:
    - Allows users to create and manage categories suited to their personal spending habits.
5. Cross-Platform Accessibility:
    - Available on both web and mobile platforms for seamless access anywhere, anytime.
6. Insightful Visualizations:
    - Graphs and charts help users quickly analyze their financial behavior.
7. Budget Tracking:
    - Helps users set monthly/weekly budgets and track spending accordingly.

Disadvantages

1. Requires Internet Connectivity:
    - Most functionalities depend on an active internet connection to sync data with the backend.
2. Limited Offline Functionality:
    - App may not offer full features in offline mode, which could be restrictive in some scenarios.
3. Learning Curve for New Users:
    - Users unfamiliar with budgeting tools may take time to fully utilize all features.
4. Dependence on External Services:
    - Email confirmations and OAuth (e.g., Google login) depend on third-party APIs which may experience downtimes.
5. Initial Data Input Overhead:
    - Users need to enter initial data manually, which might be time-consuming at first.

9. Conclusion

The **Personal Expense Tracker App** is a powerful tool designed to promote financial literacy, responsibility, and convenience among users by simplifying the process of managing daily expenses. In today's fast-paced world, where tracking money and maintaining budgets can often be overlooked, this app serves as an essential digital assistant to help individuals and businesses make informed financial decisions.

The application allows users to record their expenses, categorize them efficiently, set realistic budgets, and receive timely notifications, all through a clean and intuitive interface. It bridges the gap between manual budgeting methods and digital finance management by offering a seamless, secure, and user-friendly experience.

One of the most significant achievements of this app lies in its **real-time analysis and visual representation** of financial data. With graphs and charts, users are not only able to monitor their past and present expenses but also gain valuable insights into their spending habits. This promotes better financial planning and control.

Furthermore, the app employs modern technologies and follows full-stack development principles using the **MERN stack** (MongoDB, Express.js, React, and Node.js), ensuring both performance and scalability. Its modular architecture, reusable components, and secure authentication system ensure a robust backend and a responsive frontend, offering a reliable experience across devices.

This project also takes into account the diverse needs of its users—ranging from everyday individuals to business users—by supporting features such as multi-account management and report generation for departments or projects.

In conclusion, the **Personal Expense Tracker App** stands as a comprehensive solution to modern financial challenges. It not only meets the current needs of users but also lays a strong foundation for future enhancements such as AI-driven insights, banking integration, and predictive analytics. With the growing emphasis on personal finance management, this app has the potential to make a lasting impact in the financial technology space, empowering users to take full control of their financial well-being.

10. Future Scope

AI-Based Insights:

- Integrate machine learning to provide personalized financial advice based on user spending patterns.

Bank Account Integration:

- Link bank accounts and credit cards for automatic expense tracking and reconciliation.

Multi-Currency Support:

- Enable tracking of expenses in different currencies for international users.

Voice Assistant Integration:

- Use voice commands to log expenses or retrieve budget status.

Advanced Reporting & Forecasting:

- Predict future spending trends and savings opportunities with advanced data analytics.

Subscription Tracker:

- Track and notify users about recurring subscriptions and help manage them effectively.

Offline Mode Support:

- Allow users to log and view expenses offline with automatic syncing when online.

11. Appedix

Source Code : https://github.com/Harsh-Burman/Smartbridge-MERN-Project
Git link : https://github.com/Harsh-Burman/Smartbridge-MERN-Project
Project Demonstration:
https://drive.google.com/drive/folders/1W_GhyBoF83EJwXYNBGzqgLVz13dVFR-t?usp=sharing