

## Ideation Phase

### Brainstorm & Idea Prioritization Template

Date	15 April 2025
Team ID	SWTID1743315733
Project Name	Personal Finance Tracker
Maximum Marks	4 Marks

Here's the revised format for pasting into Word:

---

#### Brainstorm & Idea Prioritization

##### Brainstorming

During the initial planning phase for the Personal Finance Tracker, we brainstormed key features and user needs for managing personal finances effectively. The goal was to create a user-friendly tool that provides clear insights into financial health.

##### Key Ideas Generated:

- Secure user authentication (registration/login).
- Intuitive interface for adding, editing, and deleting income and expense transactions.
- Categorization of expenses (e.g., Housing, Food, Transportation) and sources for income (e.g., Salary, Freelance).
- Ability to set and track monthly budgets by category (frontend/src/pages/Budgets.js).
- Dashboard overview (frontend/src/pages/Dashboard.js) showing key stats (Total Income, Total Expenses, Balance, Savings Rate).
- Visual reports with charts (Pie chart for expenses by category, Bar chart for income vs. expenses) (frontend/src/pages/Dashboard.js, frontend/src/pages/Reports.js).
- Display of recent transactions on the dashboard.
- User profile management (frontend/src/pages/Profile.js) including password change and currency preference.
- Responsive design for usability across different devices (frontend/src/App.css).
- Filtering and pagination for transaction lists (backend/controllers/expenseController.js, backend/controllers/incomeController.js).

##### User Pain Points Identified:

- Difficulty in consistently tracking all income and expenses.
- Lack of a clear, visual overview of spending habits and financial standing.
- Challenges in setting realistic budgets and monitoring progress against them.
- Security concerns when managing financial data online.
- Existing tools often feel complex or overwhelming.

## Idea Prioritization

We used the MoSCoW prioritization technique (Must-have, Should-have, Could-have, Won't-have) to focus the development effort on core functionalities for the initial version.

Feature	Priority	Reason
User authentication	Must-have	Core security and personalized data management.
Add/Edit/Delete Expenses	Must-have	Essential for tracking spending (frontend/src/pages/Expenses.js).
Add/Edit/Delete Income	Must-have	Essential for tracking earnings (frontend/src/pages/Income.js).
Expense Categorization	Must-have	Organizes spending for analysis (backend/models/Expense.js).
Income Source Tracking	Must-have	Organizes income for analysis (backend/models/Income.js).
Budget Setup & Tracking	Must-have	Core feature for financial planning (frontend/src/pages/Budgets.js).
Dashboard Financial Summary	Must-have	Provides an immediate overview of financial health (frontend/src/pages/Dashboard.js).
Visual Reports (Charts)	Must-have	Enhances understanding of financial trends (frontend/src/pages/Dashboard.js).
Recent Transactions View	Should-have	Quick view of latest activity on the dashboard.
Profile Management	Should-have	Allows users to manage account settings (frontend/src/pages/Profile.js).
Responsive Design	Should-have	Ensures usability on various screen sizes (frontend/src/App.css).
Budget Alerts	Could-have	Future enhancement to notify users about budget limits.
Export Data (CSV/PDF)	Could-have	Adds flexibility for offline analysis or reporting.
Advanced Reporting Filters	Could-have	More granular control over report generation.
Dark Mode/Themes	Could-have	User interface customization preference.
AI Financial Suggestions	Won't-have	Complex feature requiring significant data; potential future addition.
Bank Account Integration	Won't-have	High complexity and security implications; planned for future updates.
Multi-Account Support	Won't-have	Focus on personal finance first; business/shared features later.

## Outcome of Prioritization

By prioritizing the "Must-have" and essential "Should-have" features, we focused on delivering a robust core application that allows users to effectively track income, expenses, and budgets, supported by clear visual summaries and reports. Features categorized as "Could-have" or "Won't-have" for the initial release are documented for potential inclusion in future development cycles based on user feedback and project evolution.