

Product Requirements Specification (PRS)

1. Product Overview

- **Product Name:** Money Mate
- **One-line description:** A web-based personal finance application that helps users track expenses, manage budgets, and split costs with groups.
- **Target Users:** College students, young professionals, and individuals seeking to manage personal finances and shared expenses efficiently.
- **Value Proposition:** Simplifies expense tracking and budget management while enabling transparent group expense splitting, helping users maintain financial awareness and avoid overspending.

2. Problem Statement

- Many individuals struggle to track daily expenses and stay within monthly budgets, leading to overspending and financial stress.
- Existing solutions are either too complex for casual users or lack integrated group expense management, requiring multiple platforms to track personal and shared costs.

3. Objectives

- Enable users to set and monitor monthly budgets with real-time spending updates.
- Provide clear visibility into personal expenses, savings, and amounts owed in group transactions.
- Facilitate fair and transparent splitting of group expenses among multiple participants.
- Deliver an intuitive dashboard that summarizes financial health at a glance.
- Success criteria: Users can track expenses within 2 minutes of signing up; budget alerts trigger when 80% threshold is reached; group settlements are calculated accurately with zero discrepancies.

4. Scope

- **In Scope:** User authentication, monthly budget setup, personal expense logging, group creation and expense splitting, savings tracking, owed amount calculation, monthly overview and activity summaries.
- **Out of Scope:** Investment tracking, bill payment integration, bank account synchronization, credit score monitoring, tax filing assistance, multi-currency support.

5. Core (Functional) Features

For each feature, define briefly:

- **User Authentication & Dashboard**

- **User Interaction Flow:** User registers/logs in → System validates credentials → Dashboard displays with budget overview, total spent, savings, and owed amounts.
- **AI/ML Role:** None; standard authentication and data retrieval.
- **Acceptance Criteria:** Users can register, login, and access personalized dashboard; session persists until logout.
- **Monthly Budget Setup**
 - **User Interaction Flow:** User navigates to Budget Settings → Inputs monthly budget amount → System saves and displays on dashboard.
 - **AI/ML Role:** None; direct user input storage and calculation.
 - **Acceptance Criteria:** Budget value is saved, displayed on dashboard, and used for budget utilization calculations.
- **Personal Expense Logging**
 - **User Interaction Flow:** User clicks Add Expense → Enters amount, category, description, date → System records expense and updates totals.
 - **AI/ML Role:** Rule-based categorization if category patterns are detected; optional auto-suggestion for frequently used categories.
 - **Acceptance Criteria:** Expenses are logged with timestamp; Recent Activity section updates immediately; Total Spent reflects new expense.
- **Group Expense Management**
 - **User Interaction Flow:** User creates group → Adds members → Logs group expense with split method (equal/custom) → System calculates individual shares.
 - **AI/ML Role:** None; algorithmic calculation of expense splits based on user-defined parameters.
 - **Acceptance Criteria:** Groups can be created with multiple members; expenses are split accurately; each member sees their share; settlement amounts are calculated correctly.
- **Savings Calculation**
 - **User Interaction Flow:** System automatically calculates savings as (Monthly Budget - Total Spent) → Displays on dashboard.
 - **AI/ML Role:** None; simple arithmetic calculation.
 - **Acceptance Criteria:** Savings value updates in real-time; displays correctly as budget minus expenses.
- **You Are Owed Calculation**
 - **User Interaction Flow:** System calculates net amount user is owed from group expenses → Displays on dashboard as "You Are Owed".
 - **AI/ML Role:** None; aggregation of group expense settlements where user is owed money.

- **Acceptance Criteria:** Amount accurately reflects pending settlements from all groups; updates when group expenses are added or settled.
- **Monthly Overview & Activity Summary**
 - **User Interaction Flow:** Dashboard displays budget utilization percentage, personal vs group spending breakdown, recent transactions.
 - **AI/ML Role:** Basic data aggregation and percentage calculations; optional trend visualization.
 - **Acceptance Criteria:** Monthly Overview section shows budget used percentage, breakdown of personal and group expenses; Recent Activity lists latest transactions chronologically.

6. Data Requirements

- **Data Sources:** User input via web forms (expenses, budget, group details); internally stored in application database.
- **Data Size & Quality Expectations:** Moderate volume (hundreds to thousands of transactions per user annually); data accuracy depends on user input; validation required for numerical fields.
- **Privacy / Compliance:** User data must be secured with encryption; passwords hashed; no sharing of personal financial data without consent; compliance with general data protection principles; group data visible only to group members.

7. System Behavior

- **Inputs:** User credentials, monthly budget amount, expense details (amount, category, date, description), group information (members, expense amounts, split method).
- **Outputs:** Dashboard summaries (budget, total spent, savings, owed amount), Recent Activity list, Monthly Overview with budget utilization, group expense breakdowns, settlement calculations.
- **Performance (Non-Functional) Expectations:** Page load time under 2 seconds; expense logging and dashboard updates in real-time; calculation accuracy of 100% for budget, savings, and group settlements; system availability of 99% uptime; responsive design for desktop and mobile browsers.

8. Constraints

- **Technical:** Web-based application; deployment on standard cloud hosting (e.g., Vercel, Netlify, Heroku); limited backend compute resources suitable for student project; uses relational database (e.g., PostgreSQL, MySQL) or lightweight alternatives.
- **Regulatory:** Must comply with basic data protection practices; no integration with real banking systems to avoid financial service regulations.
- **Ethical:** User expense data is private and confidential; group expense visibility limited to group members only; no selling or sharing of user financial data; transparent calculation methods for expense splitting to ensure fairness.

9. Metrics & Evaluation

- **Key evaluation metrics:** User registration and login success rate (>95%), expense logging completion rate (>90%), accuracy of budget calculations (100%), accuracy of group expense splits (100%), user satisfaction score (>7/10 in usability testing), dashboard load time (<2 seconds).
- **Target thresholds:** Must-have: 100% accuracy in calculations, functional CRUD operations for expenses and groups, secure authentication. Nice-to-have: visual charts for spending trends, export functionality for expense reports, email notifications for budget thresholds.

10. Implementation Notes (Optional but Minimal)

- **Deployment plan:** Cloud-hosted web application using React/Vue.js frontend, Node.js/Express or Django backend, PostgreSQL/MongoDB database; RESTful API for frontend-backend communication.
- **Monitoring requirements:** Error logging for failed transactions, monitoring of database query performance, tracking of user activity for feature usage insights; no model drift monitoring as no ML models are deployed.

11. Timeline & Priorities

- **Phase 1 (MVP):** User authentication, dashboard with budget cards, personal expense logging, basic savings calculation, Recent Activity display (4-6 weeks).
- **Phase 2:** Group creation and management, group expense splitting and settlement tracking, Monthly Overview with detailed breakdowns, budget threshold alerts (6-8 weeks post-MVP).

12. Open Questions

- Should users be able to export expense reports in CSV/PDF format?
- Should the system send notifications (email/push) when budget thresholds are reached?
- How should settled group expenses be archived or removed from active tracking?
- Should there be support for recurring expenses (e.g., monthly subscriptions)?
- What is the appropriate data retention policy for expense history?