MindMatrix

📱 SpendSense – Smart Expense Manager App

Comprehensive Project Requirement Document

🧭 1. Project Overview

Project Title:

SpendSense – A Smart Personal Finance & Expense Management App

Objective:

SpendSense aims to help individuals manage their daily expenses, understand their spending patterns, and make smarter financial decisions.
The app provides a clean, intuitive interface for tracking income and expenses, setting budgets, and visualizing financial data — all built using **Android Studio, Kotlin, and Jetpack Compose**.

Core Idea:

"Simplify money management for everyday users through smart insights and easy tracking."

SpendSense focuses on **data simplicity and user empowerment** — users can record their financial transactions, view spending summaries, and plan budgets effectively.

💡 2. Purpose and Goals

Purpose:

- To create a **fully functional Android app** that demonstrates the students' understanding of Kotlin, Jetpack Compose, data management, and UI design.
- To build a **real-world use case** that improves personal productivity and demonstrates strong portfolio value.

Goals:

- Track daily income and expenses.
- Help users visualize their financial health.
- Enable data-driven spending habits.
- Provide a foundation for AI-based financial predictions in the future.

MindMatrix

🎯 3. Scope of the Application

In-Scope Functionalities:

✅ Expense entry and categorization
✅ Budget creation and alerts
✅ Analytical charts and insights
✅ Local data storage with Room Database
✅ Secure user authentication (optional)
✅ Export reports (optional advanced feature)

Out of Scope (for this internship):

❌ Multi-user accounts with shared data
❌ Real-time bank API integration
❌ Cloud-based AI training models
❌ Online payment features

(These can be considered as future extensions.)

👥 4. Target Users and Personas

1. **College Students:**
   To manage pocket money, daily expenses, and savings goals.
2. **Working Professionals:**
   To track monthly spending and recurring payments.
3. **Household Budget Managers:**
   To manage family or home expenses efficiently.

User Persona Example:

Name: Priya Sharma
Age: 21
Occupation: Student
Goal: Wants to monitor her monthly spending and reduce unnecessary expenses.
Pain Points: Forgets where money goes, struggles to save.
Expectation: Wants an easy app that summarizes all her spends visually.

MindMatrix

⚙️ 5. Functional Requirements

1 User Management

- Register or login (optional for basic use).
- Store user data locally (offline mode).
- Basic profile with name, income range, and currency preference.

2 Expense & Income Management

- Add, edit, and delete expense entries.
- Categories: Food, Travel, Shopping, Rent, Health, Others.
- Each entry should include:
    - Amount
    - Category
    - Date
    - Payment Method (optional)
    - Short Note (optional)
- Add income entries to calculate monthly balance.

3 Budget Management

- User sets monthly or weekly budget.
- Real-time progress bar shows budget consumption.
- Notifications (UI-based alerts) when nearing limits.

4 Analytics Dashboard

- Display spending summary using charts.
- Graphs:
    - Category-wise spending (Pie chart).
    - Monthly trend (Bar chart).
- Summary cards showing:
    - Total Income
    - Total Expense
    - Remaining Balance

- Most spent category

5 Reports

- Generate reports by date range or category.
- Export data as CSV (optional advanced feature).

6 Smart Capabilities (Optional)

- Suggested budget goals based on history.
- Auto-category detection using simple keyword logic.
- Optional integration with Gemini API for expense analysis.

🧩 6. Non-Functional Requirements

Requirement	Description
Performance	App should load dashboards in under 2 seconds.
Usability	Intuitive, minimal UI with quick add options.
Reliability	Should not crash or lose data unexpectedly.
Portability	Compatible with Android 9.0 and above.
Scalability	Architecture should allow future AI or cloud extensions.
Security	Sensitive data (like income info) should be stored securely using Room encryption or local data protection.

🏗️ 7. System Architecture

App Architecture:

- **MVVM Pattern (Model-View-ViewModel)**
- **Local Database:** Room
- **UI:** Jetpack Compose Components
- **Data Layer:** Repository classes handle CRUD operations
- **View Layer:** Screens built using Compose

User → View (Compose UI)

↓

ViewModel

↓

Repository

↓

Room Database

🧱 8. Database Schema

Table	Fields	Description
Expense	id (PK), amount, category, date, note, payment_method	Stores expense entries
Income	id (PK), amount, date, source, note	Stores income data
Budget	id (PK), month, limit_amount	Monthly budget info
User (optional)	id (PK), name, email	Basic user data

🧭 9. User Flow Diagram (Text Outline)

1. **Splash Screen → Home Dashboard**
2. Home Dashboard → “Add Expense” or “Add Income”
3. Add Entry → Save → Return to Dashboard (update chart)
4. Menu → “Budget Settings”
5. Menu → “Reports” or “Analytics”
6. Logout / Exit (optional)

🎨 10. UI/UX Guidelines

Design Goals:

- Minimalistic and clutter-free design.
- Use neutral background (white or light gray).
- Highlight key numbers in color (green for income, red for expense).

Key Screens:

1. **Splash / Intro Screen** – App logo + tagline.
2. **Home Dashboard** – Total balance, expense summary, quick add buttons.
3. **Add Expense Screen** – Category selector + input fields.
4. **Analytics Screen** – Pie chart + monthly graph.
5. **Settings Screen** – Profile, currency, budget setup.

Color Palette: (optional to use the same)

- Primary: #1976D2 (Blue)
- Accent: #43A047 (Green)
- Warning: #E53935 (Red)
- Background: #F5F5F5 (Light Gray)

📊 11. Analytics and Reports

- Use **MPAndroidChart** or **Compose Chart Library**.
- Provide filters: date range, category, amount range.
- Display top 3 categories with highest expenses.
- Optional: Predictive chart showing next month's estimate.

🚀 12. Development Plan (6-Week Breakdown)

Week	Milestone	Deliverables
Week 1	**Project Setup & Wireframes**	App skeleton, folder structure, basic UI sketches.
Week 2	**UI Development**	Compose screens: Dashboard, Add Expense, Analytics.
Week 3	**Database Integration**	Room setup, CRUD operations for expenses/income.
Week 4	**Budget & Analytics Features**	Budget tracking, charts, and summaries.
Week 5	**Testing & Refinement**	Bug fixing, improving UI consistency.
Week 6	**Demo Preparation & Report Generation**	Final testing, video demo, and presentation.

🌎 13. Reference Apps

App	Description	Reference
Wallet by BudgetBakers	Expense & budget tracker with visual graphs.	https://budgetbakers.com
Money Manager Expense Tracker	Offline money tracker with simple UI.	Play Store
Spendee	Smart financial planner with charts and insights.	https://www.spendee.com

📋 14. Deliverables for Students

1. Wireframe (Figma/Canva)
2. User Flow Diagram
3. Functional UI Prototype
4. Final Working App (APK)
5. Project Report (features, screenshots, team roles)
6. Video Demonstration (2–3 mins)

🧠 15. Future Scope & Enhancement Ideas

- AI-based spending prediction using Gemini API.
- Cloud backup (Firebase integration).
- OCR-based receipt scanning.
- Multi-user family account mode.
- Goal-based saving planner.

MindMatrix

🧩 16. Evaluation Parameters

Category	Weightage
Functionality	30%
UI/UX Design	25%
Code Quality & Structure	20%
Innovation / Extra Features	15%
Presentation & Report	10%

✅ 17. Next Action Items for Teams

1. Brainstorm category structure and naming.
2. Prepare **wireframes and navigation flow** (to be approved by mentor).
3. Divide responsibilities — UI Developer, Database Handler, Feature Integrator.
4. Begin Week 1 setup by creating Android project in Android Studio.