

Smart Daily Expense Tracker — Full Module Assignment (AI-First)

Title

Build a Full-featured “Smart Daily Expense Tracker” Module for Small Business Owners

Context

Your app supports small business owners in digitizing their operations. Daily expenses — often unrecorded or lost on WhatsApp or paper — are a major gap in understanding cash flow. This module will help users capture, view, analyze, and export expense records easily and intelligently.

You are expected to use AI tools throughout to accelerate, generate, or optimize parts of your workflow.

Module Overview

👉 Build a multi-screen Expense Tracker module using **Jetpack Compose** (preferred) or XML-based views, with a clean **MVVM architecture**.

Required Screens & Flows

1. Expense Entry Screen

- Input fields:
 - Title (text)
 - Amount (₹)
 - Category (mocked list: Staff, Travel, Food, Utility)
 - Optional Notes (max 100 chars)
 - Optional Receipt Image (upload or mock)
- Submit Button: Adds expense, shows Toast, animates entry
- Show real-time “Total Spent Today” at top

2. Expense List Screen

- View expenses for:
 - Today (default)
 - Previous dates via calendar or filter
 - Group by category or time (toggle)
- Show: Total count, total amount, empty state

3. Expense Report Screen

- Mock report for last 7 days:
 - Daily totals
 - Category-wise totals
 - Bar or line chart (mocked)

- Export (optional):
- Simulate PDF/CSV export
- Trigger Share intent (optional)



State Management & Data Layer

- ViewModel + StateFlow (or LiveData)
- In-memory repository or Room (optional)
- Handle screen transitions via Navigation



Bonus Challenges

- Theme switcher (Light/Dark)
- Persist data locally (Room/Datastore)
- Animation on add
- Duplicate detection
- Validation (amount > 0, title non-empty)
- Offline-first sync (mock)
- Reusable UI components



AI Usage (Mandatory)

This is for AI-first Android developers. You **must** use tools like ChatGPT, Cursor, Copilot, Gemini.

Use AI for:

- UI layout ideas
- MVVM structuring
- ViewModel/Data class generation
- UX feedback and enhancements
- Prompt tuning and retries
- Code comments & README help



Deliverables

- 📁 Source Code (GitHub or Zip)
- 📄 AI Usage Summary
- 📄 APK or Download Link
- 📄 Screenshots (optional)



Submission

Include:

- 3–5 sentence AI usage summary
- Prompt logs (key prompts + retries)