1. Project Title

FinBud – A Smart Personal Finance Buddy

---

2. Problem Statement

Managing personal finances can often feel confusing and time-consuming. People struggle to track where their money goes, how much they save, and whether they're meeting their financial goals.
FinBud aims to solve this by providing an easy-to-use platform where users can record income and expenses, view visual summaries, and receive insights into their spending patterns. The goal is to empower users to make smarter financial decisions and achieve better savings habits.

---

3. System Architecture

Frontend → Backend (API) → Database

Example Stack:

Frontend: React.js + React Router + TailwindCSS + Recharts
Backend: Node.js + Express.js
Database: MongoDB Atlas (NoSQL)
Authentication: JWT-based Login/Signup
Hosting:

- Frontend → Vercel
- Backend → Render / Railway
- Database → MongoDB Atlas

Data Flow:

1. The user interacts with the frontend (dashboard, forms, charts).
2. Frontend sends API requests to backend for data (transactions, summaries, etc.).

3. Backend processes requests, interacts with MongoDB, and sends responses back.
4. Frontend displays charts and analytics in real-time.

---

4. Key Features

Category	Features
Authentication & Authorization	User registration, login, logout using JWT tokens
Transaction Management (CRUD)	Add, edit, delete, and view transactions (income/expenses)
Dashboard & Analytics	Real-time dashboard showing total income, expenses, and balance
Category-wise Charts	Visual breakdown of spending by category using Pie and Line Charts
Budget Planning	Set monthly spending limits and get alerts when nearing the budget
AI Financial Insights	Analyze user spending with OpenAI API and generate summaries like “You spent 30% more this month on food.”
Frontend Routing	Pages: Home, Login, Signup, Dashboard, Transactions, Add Transaction, Profile
Hosting	Fully deployed frontend and backend with live URLs

---

5. Tech Stack

Layer	Technologies Used
Frontend	React.js, React Router, TailwindCSS, Axios, Recharts
Backend	Node.js, Express.js
Database	MongoDB Atlas

Authentication	JWT (JSON Web Tokens)
AI	OpenAI API for personalized spending summaries
Hosting	Frontend – Vercel, Backend – Render/Railway

---

6. API Overview

Endpoint	Method	Description	Access
`/api/auth/signup`	POST	Register new user	Public
`/api/auth/login`	POST	Authenticate user	Public
`/api/transactions`	GET	Get all user transactions	Authenticated
`/api/transactions`	POST	Add a new transaction	Authenticated
`/api/transactions/:id`	PUT	Edit an existing transaction	Authenticated
`/api/transactions/:id`	DELETE	Delete a transaction	Authenticated
`/api/summary/monthly`	GET	Fetch monthly income, expense, and balance	Authenticated
`/api/summary/category`	GET	Get category-wise breakdown for charts	Authenticated
`/api/ai/insight`	POST	Generate AI-based financial summary	Authenticated