

Project Title: Welth- AI-Powered Finance Management Platform

System Design and Architecture

Introduction

The "Wealth" platform is a modular, AI-powered personal finance management system designed to automate expense tracking, budget planning, and provide AI-generated financial insights for students, young professionals, and small business owners in India. The system leverages modern web technologies, AI/ML algorithms, and OCR-based receipt scanning to enhance usability, efficiency, and security. The system design focuses on modularity, scalability, and maintainability, ensuring smooth operation even as the user base or data grows.

Modular Design

The system is divided into the following modules:

1. Front-End Module

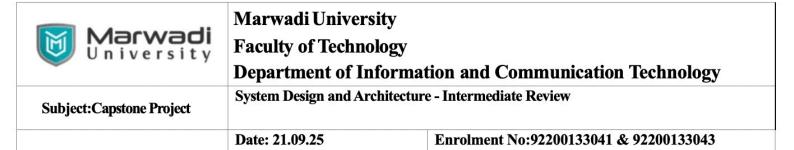
- Built using **ReactJS** for responsive UI components and smooth animations.
- Responsible for user interactions, dashboards, charts, budget visualisations, and forms.

2. Back-End Module

- Built with **Node.js** and **Express.js**.
- Handles authentication, API requests, business logic, recurring transactions, notifications, and integration with AI services.

3. Database Module

- **MongoDB** is used as a NoSQL database for flexible storage of accounts, transactions, budgets, and user profiles.
- Supports high-volume data storage, indexing, and query optimisation for fast retrieval.



Layer	Technology	Justification
Front-End	React.js, HTML, CSS	Builds dynamic UI, responsive, modern web experience
Back-End	Node.js, Express.js	Handles APIs, server logic, and communication between UI & database
Database	MongoDB (NoSQL)	Scalable, flexible schema for storing financial records & user data
AI/ML Module	Python, OCR (Tesseract)	Automates data extraction and financial insights
Notification Module	Email (SMTP / Nodemailer)	Sends alerts, reminders, and system notifications
Hosting	Vercel (Front-End), MongoDB Atlas (DB)	Free-tier deployment, easy scaling, reliable hosting

4. AI & ML Module

- **Python-based machine learning algorithms** categorise transactions and generate personalised insights.
- **OCR (Tesseract)** reads receipts and auto-fills transaction data, reducing manual entry.

Justification of Modularity:

- Each module is independent, allowing easy updates or replacements without affecting other modules.
- Improves maintainability: UI changes won't affect AI/ML logic.
- Extensible for future features like mobile apps, third-party integrations, or new analytics modules.

Technology Stack

• • • • • • • • • • • • • • • • • • •	Marwadi University		
Marwadi University	Faculty of Technology		
,	Department of Information and Communication Technology		
Subject:Capstone Project	System Design and Architecture - Intermediate Review		
	Date: 21.09.25	Enrolment No:92200133041 & 92200133043	

Rationale:

- All technologies are open-source or free for student deployment.
- Chosen for compatibility, ease of integration, and scalability potential.
- Industry-relevant stack ensures applicability for future career and maintenance.

Scalability Plan

Objective: Ensure the system can handle increased users, data, and transactions efficiently.

1. Horizontal Scaling:

- Front-end hosted on CDN-backed platforms like Vercel/Netlify for global distribution.
- Node.js back-end can be deployed across multiple instances with load balancing.

2. Database Scalability:

- MongoDB supports sharding for distributed storage across servers.
- Indexing ensures fast queries even with large datasets.

3. AI/ML Scaling:

- ML computations run asynchronously or in batch processing for multiple users.
- Future cloud-based GPU/TPU support can accelerate ML inference.

4. Caching & Performance:

- Frequently accessed data (like dashboards or recurring transactions) can be cached using Redis or in-memory storage.
- Reduces load on the database and improves response time.

5. Bottleneck Management:

• A 4	Marwadi University	
Marwadi University	Faculty of Technology	
• • • • • • • • • • • • • • • • • • • •	Department of Information and Communication Technology	
Subject:Capstone Project	System Design and Architecture - Intermediate Review	
	Date: 21.09.25	Enrolment No:92200133041 & 92200133043

Database: Sharding and indexing.

Back-End API: Load balancers and rate-limiting.

AI Processing: Asynchronous task queues (e.g., Celery, RabbitMQ).

Cost & Reliability:

- Free-tier cloud services keep costs minimal.

- Modular deployment ensures one component's failure doesn't crash the system.

- Easy to extend for premium hosting or GPU acceleration in the future.

Conclusion

The "Wealth" system's architecture is designed to be modular, scalable, and robust. Each component is clearly defined and integrates seamlessly to provide an efficient AI-powered finance platform. The chosen technology stack ensures fast development, high performance, and easy future expansion. The modular approach enhances maintainability, security, and reliability, positioning the platform for long-term adoption and success in the Indian market.