**Project Design Phase**

**Solution Architecture**

|  |  |
| --- | --- |
| Date | 15 April 2025 |
| Team ID | SWTID1743870576 |
| Project Name | SpendSmart: Your Personal Finance Companion |
| Maximum Marks | 4 Marks |

**Solution Architecture:**

### ****Solution Architecture Overview****

**Solution Architecture** is a strategic process that ensures your technology choices align effectively with business needs. For the SpendSmart project, this approach delivers a scalable, secure, and user-friendly personal finance management system.

#### ****Purpose of Solution Architecture in HOUSE HUNT****

* **Identify Best Tech Solution:** Leverage the MERN stack (MongoDB, Express.js, React.js, Node.js) to build a responsive, real-time web application for personal expense tracking and analytics.
* **Communicate with Stakeholders:** Visualize and communicate how users interact with the system, what financial data is processed, and how the backend supports secure, efficient operations.
* **Define Features & Phases:** Clearly structure the development timeline through sprints—starting from user registration and authentication to transaction management, analytics, and dashboard features.
* **Deliver Specifications:** Provide technical documentation including RESTful APIs, data models, authentication mechanisms, and UI/UX workflows.

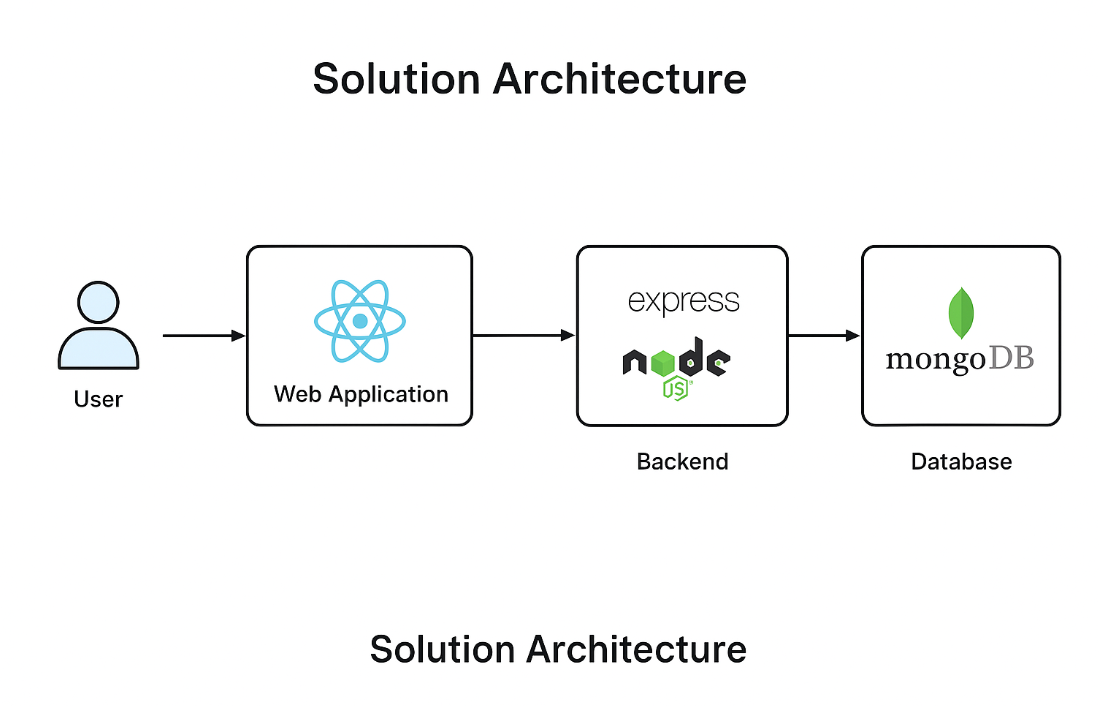
### ****Key Components of the Solution Architecture****

| **Component** | **Description** |
| --- | --- |
| **Frontend (React.js)** | Delivers a dynamic, responsive, and intuitive UI for users to add, view, and categorize income and expenses. |
| **Backend (Node.js + Express.js)** | Handles API routing, business logic, transaction management, and user authentication. |
| **Database (MongoDB)** | Stores user profiles, categorized transactions, and financial summaries securely in a flexible NoSQL format. |
| **Authentication** | JWT-based secure login and signup for users, ensuring privacy and data protection. |
| **Analytics & Visualization** | Generates real-time charts and summaries of income and expenses using Chart.js for actionable financial insights. |
| **Search & Filters** | Allows users to filter transactions by date, category, or amount for easy tracking and review. |
| **Dashboard Module** | Central hub displaying key financial metrics, recent transactions, and spending trends at a glance. |
| **Deployment** | Hosted on cloud platforms (like Vercel/Netlify for frontend, and Render/Heroku for backend) for high availability and scalability. |

### ****Development Phases****

1. **Sprint 1:** User registration, login, and secure authentication (JWT).
2. **Sprint 2:** Transaction management (add, edit, delete, categorize income/expenses), integration with MongoDB.
3. **Sprint 3:** Analytics and visualization (charts, summaries), dashboard UI, responsive design.
4. **Sprint 4:** Final integration, testing, performance optimization, and deployment.

**Example - Solution Architecture Diagram:**

****

*Figure 1: Architecture and data flow of the voice patient diary sample application*

**Reference:** [**https://aws.amazon.com/blogs/industries/voice-applications-in-clinical-research-powered-by-ai-on-aws-part-1-architecture-and-design-considerations/**](https://aws.amazon.com/blogs/industries/voice-applications-in-clinical-research-powered-by-ai-on-aws-part-1-architecture-and-design-considerations/)