**Requirement Specification Document for Smart Financial Manager System**

**1. Introduction**

The **Smart Financial Manager System** is envisioned as a comprehensive platform to assist users in managing their personal finances. The system will provide features for tracking expenses, setting budgets, monitoring savings, and generating insightful financial reports. This document outlines the requirements gathered through various techniques such as observations, surveys, and questionnaires, aimed at understanding customer needs and ensuring the development of a user-centric system.

**2. Requirement Gathering Techniques**

**2.1 Observation**

* Observed individuals using spreadsheets and manual methods to track their finances.
* Identified common pain points such as difficulty in organizing data, lack of automation, and errors in calculations.

**2.2 Surveys**

* Conducted surveys among 200 potential users to gather insights about their financial management habits.
* Key findings:
  + 75% of respondents struggle to stay within their budgets.
  + 60% wish to automate tracking of expenses and savings.
  + 50% want personalized recommendations for better financial planning.

**2.3 Questionnaires**

* Distributed questionnaires to a focus group of 50 individuals, asking about:
  + Desired features in a financial management system.
  + Challenges they face with existing tools.
  + Preferences for user interface and experience.

**2.4 Interviews**

* Conducted one-on-one interviews with financial experts and frequent users of budgeting tools.
* Extracted valuable input on features such as reports, goal tracking, and security.

**3. Functional Requirements**

**3.1 User Management**

* **Registration**: Users must be able to create accounts using email and password.
* **Login**: Users must log in securely with authentication mechanisms.
* **User Roles**: Basic users, premium users with access to advanced features.

**3.2 Financial Management**

* **Year and Month Selection**: Users must first select the year and month for which they want to manage their records.
* **Budget Management**: Users can set a monthly budget for the selected year and month.
* **Expense Tracking**: After setting a budget, users can add and categorize expenses.
* **Savings Goals**: Users can set and monitor savings goals for the selected period.

**3.3 Reporting**

* **Financial Reports**: Generate visual and textual reports summarizing expenses, budgets, and savings.
* **Insights and Alerts**: Provide recommendations and alerts when spending exceeds budgets.

**3.4 Security**

* Data encryption for secure storage.
* JWT-based authentication for session management.

**4. Non-Functional Requirements**

* **Usability**: Intuitive and user-friendly interface.
* **Performance**: Fast response times, even with large datasets.
* **Scalability**: Support up to 10,000 users initially, scalable as needed.
* **Accessibility**: Compatible with web and mobile platforms.
* **Availability**: 99.9% uptime.
* **Localization**: Multi-language support.

**5. Use Case Scenarios**

**Use Case 1: User Registers and Logs In**

* **Actor**: User
* **Precondition**: User visits the website.
* **Steps**:
  1. User clicks on "Register."
  2. User provides username, email, and password.
  3. System confirms registration.
  4. User logs in with credentials.
* **Postcondition**: User is redirected to the dashboard.

**Use Case 2: User Selects Year and Month**

* **Actor**: User
* **Precondition**: User is logged in.
* **Steps**:
  1. User navigates to the "Year and Month" selection screen.
  2. User selects the desired year and month from dropdowns.
  3. System saves the selection and updates the dashboard.
* **Postcondition**: User's selection is set for further operations.

**Use Case 3: User Sets a Budget**

* **Actor**: User
* **Precondition**: User has selected a year and month.
* **Steps**:
  1. User navigates to "Set Budget."
  2. User specifies the monthly budget for the selected year and month.
  3. System saves the budget and provides feedback on spending.
* **Postcondition**: Budget is updated and visible on the dashboard.

**Use Case 4: User Adds an Expense**

* **Actor**: User
* **Precondition**: User has set a budget for the selected year and month.
* **Steps**:
  1. User navigates to "Add Expense."
  2. User fills out expense details (amount, category, reason, date).
  3. System saves the expense and updates the dashboard.
* **Postcondition**: Expense is added successfully.

**6. Identified Areas for Further Research**

**6.1 User Needs**

* More research is needed to understand preferences for report customization and visualization.
* Investigate the demand for AI-driven financial advice.

**6.2 Technical Constraints**

* Assess database options to support scalability and performance.
* Determine the best approach for real-time data synchronization.

**7. Basic System Objectives**

* Provide users with an intuitive tool to manage personal finances effectively.
* Enable seamless tracking of budgets, expenses, and savings goals.
* Enhance financial decision-making through insightful reporting and alerts.
* Ensure data security and user privacy.

**8. Conclusion**

The requirements outlined in this document provide a foundation for the development of the Smart Financial Manager System. By prioritizing user needs and leveraging feedback from diverse sources, the system aims to become an indispensable tool for effective financial management. Future iterations of this document will refine the requirements based on ongoing research and stakeholder input.