### **Business Requirements Document (BRD)**

**Project Name:** SNEA (Simple Network Engine Assistant)  
 **Version:** 1.0  
 **Date:** April 14, 2025  
 **Author:** Sneha AJITHKUMAR  
 **Status:** Draft

### **1. Executive Summary**

**SNEA** is a conversational AI chatbot designed to assist users in obtaining banking information and support. The chatbot will integrate with Firebase for backend operations and Dialogflow for natural language processing (NLP), providing a user-friendly interface that allows customers to interact with the banking system. This personal project will follow agile development principles with an MVP delivered within one week.

### **2. Project Overview**

#### **2.1 Background**

With the increasing demand for automated customer service in the banking sector, conversational AI solutions like HDFC EVA have proven effective in improving customer experience. **SNEA** aims to replicate this success on a smaller scale while allowing the developer (you) to learn and apply Firebase, Dialogflow, and n8n technologies in practice.

#### **2.2 Objectives**

* Develop an AI-powered chatbot to handle basic banking queries and simulate interactions like balance inquiries, product details, and customer support.
* Integrate core technologies (Firebase, Dialogflow, n8n) to provide a seamless experience.
* Deliver a minimum viable product (MVP) that can be iterated upon.
* Follow software development lifecycle practices and document the process.

#### 

#### **2.3 Success Criteria**

* **Functionality:** SNEA responds to at least 90% of test queries with relevant, accurate answers.
* **UI:** The user interface is responsive and accessible across devices.
* **Core Features:** Basic banking information and FAQs are addressed effectively.
* **Timeline:** Project is completed within 7 days.
* **Documentation:** The project is documented for future enhancements.

#### **2.4 Scope**

**In Scope:**

* Conversational AI powered by Dialogflow and Firebase.
* Core banking information including FAQs about accounts, loans, credit cards, and service processes.
* Simulated user queries and responses for banking services.
* Workflow automation using n8n for potential future enhancements.
* Code and documentation for the project.

**Out of Scope:**

* Integration with actual banking systems (APIs, credentials, etc.).
* Real authentication or personal account management.
* Full-scale security measures and production-level deployment.
* Development of mobile apps (the solution will be web-based only).
* Multi-language support (English only in the MVP).

### **3. Stakeholders**

#### **3.1 Primary Stakeholder**

* **Sneha AJITHKUMAR** (Project Owner & Developer)

#### **3.2 Secondary Stakeholders**

* **Ajithkumar KUMARAN** (Potential Review Stakeholder)
* **End Users** (Bank customers using the chatbot)
* **Developers** (Others who may replicate or extend the project)
* **Portfolio Reviewers** (For demonstrating the project as part of your personal portfolio)

### **4. Functional Requirements**

#### **4.1 Conversation Capabilities**

##### **4.1.1 Basic Conversation**

* **Greeting and Introduction:** The chatbot greets the user and provides a brief overview of its capabilities.
* **Small Talk:** The chatbot can engage in small talk with users (e.g., "How can I assist you today?" or "How's the weather?").
* **Help & Guidance:** The chatbot will offer guidance on available services and functions.
* **Multi-turn Context:** The chatbot maintains conversation context to handle multi-turn dialogues (e.g., asking follow-up questions about services).

##### **4.1.2 Banking Information**

* **Product Details:** SNEA will provide information about various banking products such as savings accounts, loans, and credit cards.
* **FAQ Handling:** SNEA will be able to respond to commonly asked questions, such as interest rates, loan processes, account opening, and other banking procedures.
* **Fees & Charges:** The bot will explain the fees and charges associated with banking products and services.
* **Interest Rates:** SNEA will provide information about interest rates for various banking products.

##### **4.1.3 Simulated Account Information**

* **Balance Inquiry:** The chatbot will respond to queries about account balance (simulated data for demo purposes).
* **Recent Transactions:** Users can inquire about recent transactions (simulated).
* **Account Statements:** SNEA will provide simulated account statement details.

##### **4.1.4 Service Information**

* **Banking Product Applications:** SNEA will explain how users can apply for different banking products like loans or credit cards.
* **Documentation Requirements:** The bot will provide a list of required documents for various banking services.
* **Branch/ATM Locations:** SNEA will share predefined information about the locations of nearby bank branches and ATMs.
* **Customer Service Details:** The chatbot will provide contact details for customer support.

#### **4.2 User Experience**

##### **4.2.1 Interface Requirements**

* **User-Friendly Interface:** A clean, intuitive chat interface with quick-reply buttons for common actions.
* **Structured Information:** Use of cards to display structured data (e.g., product information, recent transactions).
* **Responsive Design:** The chatbot interface will be responsive across all devices, including desktop and mobile.

##### **4.2.2 Conversation Flow**

* **Natural Progression:** The chatbot will follow a natural conversational flow, adapting to user input.
* **Context Retention:** The chatbot will remember previous exchanges and context for a smoother experience.
* **Fallback Responses:** If the chatbot cannot understand a user query, it will return a helpful fallback response or ask for clarification.
* **Restart Conversation:** Users can restart the conversation at any time.

### **5. Technical Requirements**

#### **5.1 Platform Requirements**

##### **5.1.1 Development Technologies**

* **Firebase:** For backend hosting, authentication, and real-time Firestore database.
* **Dialogflow ES:** For natural language processing and intent management.
* **HTML/CSS/JavaScript (React optional):** Frontend development for a chat interface.
* **n8n:** Workflow automation for advanced functionalities (e.g., ticketing, notifications).
* **Git/GitHub:** Version control for the codebase.

##### **5.1.2 Integration Points**

* **Dialogflow API:** Integrated with Firebase to handle user queries.
* **Firebase SDK:** Integrated with the frontend for real-time interaction.
* **n8n Webhooks:** For automating workflows triggered by specific actions (e.g., sending tickets or notifications).

#### 

#### **5.2 Security Requirements**

##### **5.2.1 Data Protection**

* **Simulated Data:** Only non-sensitive, simulated data will be used.
* **Secure Storage:** Conversations and logs will be stored securely in Firestore.

##### **5.2.2 Authentication**

* **Basic Firebase Authentication:** Used for any basic authentication in the demo (i.e., optional login system for advanced features).

#### **5.3 Performance Requirements**

##### **5.3.1 Response Time**

* **Response Delivery:** The chatbot should respond to user queries within 2 seconds.

##### **5.3.2 Availability**

* **Web Accessibility:** The application will be available through modern web browsers, optimized for mobile devices.

### **6. Implementation Approach**

#### **6.1 Development Methodology**

* **Agile Development:** Modular, iterative approach with focus on daily progress and regular testing.
* **Sprints:** Work will be broken down into small modules, with one feature developed, tested, and deployed before proceeding to the next.

#### **6.2 Testing Strategy**

* **Unit Testing:** Core functionalities (e.g., conversation flow, Dialogflow intents).
* **User Acceptance Testing (UAT):** Validating chatbot responses against predefined user queries.
* **End-to-End Testing:** Ensuring all integrations (Firebase, Dialogflow, n8n) function smoothly together.

#### **6.3 Timeline**

* **Planning & Requirements:** 1-2 days
* **Design & Architecture:** 1 day
* **Implementation:** 3-4 days
* **Testing & Deployment:** 1 day

### **7. Constraints and Assumptions**

#### **7.1 Constraints**

* **Limited Timeline:** One-week timeline for MVP development.
* **Basic Setup:** Using "vibe coding" approach, not full-stack development.
* **Learning Focus:** This is a learning project, so certain optimizations may not be production-ready.

#### **7.2 Assumptions**

* **Free Tier Usage:** Firebase and Dialogflow will be used within their free tiers.
* **Simulated Data:** No real banking system integration or sensitive data will be involved.

### **8. Risks and Mitigations**

#### **8.1 Risks**

* **Limited Experience:** Development may take longer due to lack of full-stack knowledge.
* **Integration Issues:** Integrating Firebase, Dialogflow, and n8n may require debugging.

#### **8.2 Mitigation Strategies**

* **Prioritize Core Features:** Focus on the most essential features for the MVP.
* **Follow Tutorials:** Rely on online documentation and resources for Firebase, Dialogflow, and n8n.

### **9. Appendices**

#### **9.1 Glossary**

* **SNEA:** Simple Network Engine Assistant
* **NLP:** Natural Language Processing
* **Intent:** The purpose behind a user's input (e.g., asking for a loan product).
* **Entity:** Specific pieces of information extracted from user input (e.g., loan amount).
* **Webhook:** A real-time callback mechanism to integrate systems.

#### **9.2 References**

* **Firebase Documentation**
* **Dialogflow ES Documentation**
* **n8n Documentation**
* **HDFC EVA Public Info**