BRD Analysis - 2025-08-12

Okay, thank you for providing the full BRD content. This is excellent. Let’s start by analyzing it. **Initial Assessment:** The BRD is fairly comprehensive, covering key aspects of the IDFC Gen-AI App Store API. However, there are areas that could be strengthened for clarity and completeness. The extensive use of example cURL commands is helpful, but it could be better integrated within the documentation. **Key Strengths:** * **Detailed Functionality Coverage:** The BRD clearly outlines the various use cases supported by the API (document analysis, RAG, classification, etc.). * **Model Options:** The inclusion of different AI model options with descriptions and "best for" recommendations is a valuable feature. * **Error Handling:** The detailed error handling section with specific status codes and solutions is crucial for developers. * **Security Features:** The guardrail system and its functionalities are well-defined. **Areas for Improvement & Questions:** 1. **Performance Metrics - Quantifiable Goals:** While the BRD mentions “96.7% success rate,” it lacks specific, measurable performance goals. What’s the target for response times? Are there Service Level Agreements (SLAs) in place? 2. **RAG Implementation - Chunking Strategy:** The BRD mentions "chunking," but it doesn’t detail the chunking strategy. How are documents segmented? What is the average chunk size? This is critical for understanding RAG performance. 3. **Guardrail System - Granularity:** Can you elaborate on the ‘safe’ classification? What level of sensitivity does the guardrail system detect (e.g., profanity, specific keywords relating to financial risks)? 4. **Model Comparison - Quantitative Differences:** The descriptions of the AI models (e.g., “simplyfillm runs all models…”) are good, but could benefit from quantifiable differences. For example, what’s the expected latency difference between ‘gemma3:4b’ and ‘gemma3:12b’ for a given task? 5. **Flowchart for RAG Implementation**: I would like to request the creation of a flowchart to visually illustrate the RAG implementation process. 6. **Missing Diagram**: There is no diagram of the overall architecture. It would be extremely useful to visualize how the API integrates with other systems. 7. **Non-Functional Requirements - Scalability**: Does the API have scaling capabilities to handle potentially increasing loads or an increasing number of users? **Next Steps:** To help me assist you further, could you tell me: * What specific aspect of the API are you most interested in focusing on right now (e.g., RAG implementation, error handling, model selection)? * Would you like me to start drafting a basic architecture diagram, or perhaps a flowchart illustrating the RAG process?