DrugBot

AI-Powered Pharmaceutical Assistant

Project	DrugBot – Gemini 3 Hackathon
Model	Gemini 3 Flash Preview
Protocol	Model Context Protocol (MCP)
Version	1.1
Date	Feb-26

1. Objective

This document defines the business requirements for DrugBot, an AI-powered Pharmaceutical Assistant developed for the Gemini 3 Hackathon. DrugBot leverages the Model Context Protocol (MCP) to connect a Gemini 3 model to live FDA data sources, bridging the gap between complex raw regulatory data and actionable consumer safety information.

The system provides a unified chat interface for querying real-time pharmaceutical data including adverse events, product labeling, recall enforcement reports, and drug shortage information directly from official FDA databases.

2. Target Users

The system serves the following user personas, with patients as the primary audience:

Persona	Description	Primary Needs
Patients (Primary)	Individuals managing their own medications and health decisions.	Check drug safety, verify recalls, understand side effects, find alternatives during shortages.
Healthcare Providers	Physicians, pharmacists, and nurses making prescribing decisions.	Quick lookup of prescribing information, recall status, and shortage data during clinical workflows.
Caregivers	Family members or professionals managing medication for dependents.	Monitor safety alerts and shortages for medications administered to others.

3. Core Objectives & Functional Requirements

3.1 Centralized Access to Critical FDA Data

The system must provide a unified interface for four essential categories of pharmaceutical information:

- **Adverse Events:** Patient safety reports and side-effect profiles to help users understand potential risks associated with specific medications.
- **Product Labeling:** Official prescribing information, including dosage guidelines, warnings, contraindications, and drug interaction data.
- **Recall Enforcement:** Real-time monitoring of drug recalls (Class I, II, and III) to alert users of dangerous or defective products currently on the market.
- **Drug Shortages:** Current supply availability status to help patients and providers navigate medication scarcities and identify alternatives.

3.2 Intelligent Context Handling (The MCP Advantage)

Unlike standard chatbots, this assistant utilizes MCP Streamable HTTP transport to connect the Gemini 3 model to live FDA data sources, enabling an agentic workflow with the following requirements:

- **Proactive Safety:** The system must automatically offer follow-up safety checks. For example, after providing dosage information for a drug, it must proactively ask the user if they would like to check for recent recalls or shortages of that specific drug.
- **Value-First Interaction:** If a user asks a broad question (e.g., "Are there any recalls?"), the system must immediately invoke the appropriate tool (e.g., get_critical_recalls) to provide instant, actionable value rather than asking for clarification first.
- **Contextual Continuity:** The system must maintain conversation context across multiple turns so that follow-up questions (e.g., "What about side effects?") are resolved against the previously discussed drug without requiring the user to restate it.

3.3 High-Fidelity Data Presentation

Regulatory data must be presented accurately and completely within reasonable operational limits:

- **Complete Results for Bounded Queries:** The UI must display all items returned by the FDA database for standard queries (e.g., all 25 active shortages for a drug). For queries that return exceptionally large result sets (500+ items), the system must implement pagination or progressive loading with a summary, ensuring no critical data is silently dropped.
- **Structured Reporting:** Every recall entry must include the specific reason for recall (e.g., "microbial contamination"), affected lot numbers, classification level, and geographic distribution.
- **Source Attribution:** All data presented must include a reference to the FDA source endpoint and the date of retrieval to maintain auditability.

4. Non-Functional Requirements

4.1 Performance

- **Response Latency:** Tool calls to the FDA API should resolve within 5 seconds under normal conditions. The UI must display a loading indicator for any request exceeding 2 seconds.
- **Concurrent Users:** The system must support at least 10 concurrent sessions for demo purposes without degradation.

4.2 Reliability & Error Handling

- **API Unavailability:** If the openFDA API is unreachable or returns an error, the system must display a clear, user-friendly message (e.g., "FDA data is temporarily unavailable. Please try again shortly.") rather than failing silently or showing raw error output.
- **Rate Limiting:** The MCP server must implement basic rate-limiting awareness to avoid exceeding openFDA API quotas, queuing or throttling requests as needed.
- **Graceful Degradation:** If one data category (e.g., shortages) is unavailable, the remaining categories must continue to function normally.

4.3 Data Freshness

All queries are executed live against the openFDA API with no caching layer for the hackathon demo. This ensures data freshness at the cost of higher latency. A future production version may introduce a caching strategy with defined TTLs per data category.

5. System Architecture

The project follows a modular three-tier architecture to ensure separation of concerns, independent scalability, and ease of deployment:

Component	Technology	Responsibility	Deployment
MCP Server	Python / FastMCP	Data gatekeeper: fetches, filters, and optimizes FDA API responses for LLM consumption.	Google Cloud MCP
MCP Client	FastAPI / Gemini SDK	Orchestration layer: manages the agentic workflow, tool execution, session state, and conversation history.	Google Cloud MCP
UI	Next.js	Presentation layer: modern, responsive chat interface for end-user interaction with the AI assistant.	Vercel

The MCP Server and Client are deployed to Google Cloud.

6. Success Metrics

The following metrics define a successful hackathon demonstration:

Metric	Target	Measurement
Recall Identification Speed	User can identify a recalled drug in under 30 seconds from first message.	Timed demo scenario
Proactive Safety Trigger Rate	System offers follow-up safety checks in 100% of drug-specific queries.	Conversation log audit
Data Completeness	Zero truncated results for queries returning fewer than 100 items.	Comparison against direct API calls
End-to-End Availability	All four data categories operational during live demo.	Pre-demo checklist verification

7. Business Impact

By automating the retrieval and simplification of FDA data, DrugBot delivers the following value:

- **Patient Safety:** Reduces the risk of medication errors by making adverse event data and drug interactions accessible in plain language.
- **Recall Transparency:** Provides immediate visibility into active drug recalls, enabling patients and providers to act before harm occurs.
- **Supply Chain Awareness:** Empowers patients and pharmacists with real-time shortage data to proactively find alternatives rather than discovering unavailability at the pharmacy counter.

8. Assumptions & Constraints

- The openFDA API is available and responsive during the hackathon demo window.
- The Gemini 3 Flash Preview model is accessible via the Google GenAI SDK with a valid API key.
- In-memory session storage is acceptable for the demo; persistent storage is out of scope.

- The system is not intended to provide medical advice and will include appropriate disclaimers.
- Data accuracy is dependent on the openFDA API; DrugBot does not independently verify FDA data.