Requirement Analysis

Solution Requirements (Functional & Non-functional)

Date	February 18, 2026
Team ID	LTVIP2026TMIDS80425
Project Name	Smart Bridge – Intelligent SQL Querying
Maximum Marks	4 Marks

Functional Requirements

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Natural Language Query Input	• Text input field for natural language questions • Submit button to trigger query processing • Input validation and sanitization
FR-2	AI-Powered SQL Generation	• Integration with Google Gemini API • Schema-aware prompt construction • SQL query extraction from AI response
FR-3	Query Execution Engine	• Execute generated SQL on SQLite database • Return structured results (columns + rows) • Error handling for invalid SQL
FR-4	Database Schema Viewer	• Display all tables in the database • Show columns with data types for each table • Interactive expandable/collapsible table list
FR-5	Results Display	• Display query results in formatted table • Show generated SQL in code block • Display AI explanation of the query
FR-6	Sample Database	• Seed script for sample e-commerce data • Tables: customers, products, orders, order_items • Realistic sample data for demonstration

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-7	RESTful API	• POST /api/query – process natural language query • GET /api/schema – retrieve database schema • CORS support for frontend communication

Non-functional Requirements

Following are the non-functional requirements of the proposed solution.

NFR No.	Non-Functional Requirement	Description
NFR-1	Usability	The application shall provide an intuitive, modern dark-themed UI with clear input fields, readable results tables, and a schema sidebar that is easy to navigate for non-technical users.
NFR-2	Security	The system shall only allow read-only SQL operations (SELECT queries). All user inputs shall be sanitized to prevent SQL injection. API keys shall be stored securely in environment variables.
NFR-3	Reliability	The system shall handle API failures gracefully and provide meaningful error messages. The application shall be available 99% of the time during normal operating conditions.
NFR-4	Performance	The system shall respond to natural language queries within 5 seconds (including AI processing). Database queries on the sample dataset shall execute in under 100ms.
NFR-5	Availability	The backend API shall be accessible via HTTP on configurable port. The frontend shall be served as a static web application accessible via any modern browser.
NFR-6	Scalability	The architecture shall support swapping the database engine (SQLite to PostgreSQL/MySQL). The AI service layer shall be modular to support different LLM providers.