

**Project Design Phase-II**  
**Solution Requirements (Functional & Non-functional)**

Date	19 January 2025
Team ID	LTVIP2026TMIDS76399
Project Name	IntelliSQL: Intelligent SQL Querying with LLMs Using Gemini Pro
Maximum Marks	4 Marks

**Functional Requirements:**

FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	The system should provide a simple and user-friendly interface where users can easily enter natural language queries and view results without technical knowledge of SQL.
NFR-2	<b>Security</b>	User data and database credentials must be protected using authentication, encrypted communication (HTTPS), and secure API handling while interacting with Gemini Pro
NFR-3	<b>Reliability</b>	The system should consistently generate correct SQL queries and handle errors gracefully without crashing, even for complex or invalid user inputs.

NFR-4	<b>Performance</b>	The system should convert natural language to SQL and display results within a few seconds to ensure smooth user experience.
-------	--------------------	--

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIn
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Natural Language Query Processing	User enters query in natural language Send query to Gemini Pro LLM Convert natural language into SQL query Validate generated SQL syntax
FR-4	Natural Language Query Processing	Execute generated SQL query on database Fetch results from database Display results in tabular format Show error message for invalid queries

#### Non-functional Requirements:

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

NFR-5	<b>Availability</b>	The application should be accessible 24/7 with minimal downtime, ensuring users can query the database anytime.
NFR-6	<b>Scalability</b>	The system should support multiple users simultaneously and handle increasing database size or query load without performance degradation.