

User Acceptance Testing (UAT) Template

| | |
|---------------|---|
| Date | 08 February 2026 |
| Team ID | LTVIP2026TMIDS44479 |
| Project Name | Intelligent SQL Querying with LLMs Using Gemini Pro |
| Maximum Marks | |

Project Overview:

Project Name: Intelli SQL – Intelligent SQL Querying with LLMs Using Gemini Pro

Project Description:

Intelli SQL is an AI-powered system that converts natural language queries into optimized SQL queries using Gemini Pro. The system allows users to interact with databases without deep SQL knowledge, enabling efficient data retrieval, query execution, and performance monitoring through dashboards.

Project Version: 1.0

Testing Period: 15 February 2026 to 20 February 2026

Testing Scope:

The scope of User Acceptance Testing includes end-to-end validation of IntelliSQL functionalities.

Model Functionality Validation

- Verifying natural language to SQL conversion accuracy
- Testing SQL execution on SQLite database
- Validating confidence score generation

User Interface & Workflow

- Testing query input field
- Verifying SQL output display
- Checking result table formatting

Dashboard & Reports

- Validating Power BI / Tableau dashboards
- Checking dynamic filter updates
- Verifying KPI metrics (Accuracy, Response Time)

Error Handling

- Testing empty input
- Testing invalid queries
- Testing malicious SQL input

Performance Validation

- Measuring response time
- Testing multiple query submissions
- Checking system stability under moderate load

Testing Environment:

URL/Location: http://localhost:8501

Credentials (if required):

Username – test_user

Password – Test@123

Test Cases:

| Test Case ID | Test Scenario | Test Steps | Expected Result | Actual Result | Pass/Fail |
|--------------|------------------------------------|--|-------------------------------|-----------------------------------|-----------|
| TC-001 | Generate SQL from Natural Language | 1. Open app2. Enter query3. Click Generate | Correct SQL query generated | SQL generated correctly | Pass |
| TC-002 | Execute Generated SQL | 1. Generate SQL2. Execute query | Results displayed correctly | Correct results returned | Pass |
| TC-003 | Test Invalid Input | 1. Leave input blank2. Click Generate | Show error message | “Input cannot be empty” displayed | Pass |
| TC-004 | SQL Injection Prevention | 1. Enter malicious SQL2. Execute | System blocks unsafe query | Unsafe query rejected | Pass |
| TC-005 | Dashboard Filter Check | 1. Open dashboard2. Apply filter | Dashboard updates dynamically | Filters working correctly | Pass |
| TC-006 | Response Time Validation | 1. Submit query2. Measure time | Response < 5 seconds | Avg 2.8 seconds | Pass |
| TC-007 | Low Confidence Warning | 1. Enter ambiguous query | System flags low confidence | Warning displayed | Pass |

Bug Tracking:

| Bug ID | Bug Description | Steps to Reproduce | Severity | Status | Additional Feedback |
|--------|-----------------|--------------------|----------|--------|---------------------|
|--------|-----------------|--------------------|----------|--------|---------------------|

| | | | | | |
|---------------|------------------------------|---------------------------|-----------------|---------------|-----------------------------|
| BG-001 | Minor delay in JOIN queries | Submit complex JOIN query | Medium | Closed | Optimized prompt structure |
| Bug ID | Bug Description | Steps to Reproduce | Severity | Status | Additional Feedback |
| BG-002 | Dashboard slow on large logs | Load large dataset | Medium | In Progress | Implement caching mechanism |

Sign-off:

Tester Name: Reddy Manoj Kumar
Date: 21 February 2026 Signature:
R. Manoj Kumar

Notes:

- Both positive and negative scenarios were tested.
- Error handling and security validation were verified.
- System performance is stable and reliable.
- Ready for deployment after minor performance optimizations.