



# AI QA Automation Engine – Jira + Google Sheets + SQL Integration

## Overview:

This project is a fully automated QA pipeline designed for real-world software testing and reporting workflows. It integrates JIRA API, Google Sheets, and SQLite, with automated log reporting, error tracking, and evaluation metrics — all inside a single reproducible Jupyter/Colab notebook.

## Features:

AI Response Validation Engine — Automatically tests generated responses and logs pass/fail results.

- JIRA Automation — Creates, updates, and links bug reports directly from failed test logs.
- Google Sheets Integration — Syncs QA results for management visibility and collaboration.
- SQLite Database — Stores persistent logs and error history for reproducibility.
- Evaluation Metrics — Computes accuracy, precision, and fail trends.
- Daily Log Reports — Generates new per-day CSV and updates cumulative logs.
- Pytest Integration — Allows structured test cases to run against AI outputs.



**Latest Update — LLM Integration (v2.5 Final)**

## New Additions:

- Integrated dual-mode testing system:
  - Gemini 2.5 (Real LLM) — used for real-time QA practice with free API testing.
  - Invisible AI (Dummy LLM) — pre-configured structure for company SDK integration upon hiring.

## Features Added:

- 1) Dynamic API Switching: Easily toggle between Gemini and Invisible AI blocks for training or production use.
- 2) Fuzzy Evaluation Logic: Added containment-based text normalization (`normalize_text()`) to handle descriptive or semantically correct answers from LLMs.
- 3) Fully Automated Logging: Real-time CSV generation and version comparison across multiple runs.

#### 4) Pytest Integration Updated: Enhanced with normalized string comparison for more accurate QA results.

### Outcome:

The notebook now functions as a ready-to-deploy QA Automation Engine, capable of validating AI model responses from any API-driven LLM. It is designed for both enterprise QA testing (Invisible AI) and individual practice (Gemini 2.5).

### Architecture:

Data Layer: CSV / SQLite for QA logs.

Cloud API: Google Sheets API for syncing reports.

Issue Tracker: JIRA REST API for automated ticket creation.

Runtime: Jupyter / Colab.

Automation: Python (requests, pandas).

### Setup:

1. **JIRA API Token:** Create an API token via Atlassian Account > Security > Create API Token.
2. **Google Sheets API:** Enable Sheets & Drive APIs in Google Cloud Console.
3. **SQLite:** QA logs automatically sync to /content/drive/MyDrive/AI\_QA\_Logs/qa\_log.db.

### Pipeline Flow:

1. AI generates response
2. Evaluation metrics run → pass/fail decided
3. Failures logged into error\_log\_report.csv
4. JIRA API auto-creates or updates issues
5. Google Sheets syncs daily summaries
6. SQLite stores data for reproducibility

## **Tech Stack:**

Language: Python 3.x

Libraries: pandas, requests, sqlite3, gspread, pytest

Environment: Google Colab / Jupyter Notebook

External APIs: JIRA REST API v3, Google Sheets API

## **Adversarial & AI Safety Testing:**

Built an LLM Red-Teaming system to evaluate robustness against jailbreaks, bias, and toxicity. Logged and visualized results, and integrated failure reports into JIRA for continuous monitoring.

## **Author:**

Tawhidul Hasan Shanto ( Shan)

Automated QA & AI Workflow Engineer

- [tawhidshan37@atlassian.net](mailto:tawhidshan37@atlassian.net)  
[tawhidulhasanshan@gmail.com](mailto:tawhidulhasanshan@gmail.com)

## **License:**

This project is open for educational and non-commercial purposes. Feel free to modify for QA automation learning use cases.