iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 2026 **Issue:** Conference **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.83180

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com
National Conference on Integrated Approaches to Sustainable Innovation 2026 (IASI 2026)

Enhancing Academic Evaluation: A Modern GUI-Based Quiz Application for Local Environments

Anushka Mukherjee[1], Dhrubajyoti Ghosh[2]
[1, 2]*Department Of Computer Science & Engineering, Om Dayal Group Of Institutions, Uluberia, Howrah*

Abstract: ***This paper presents the design and implementation of a Graphical User Interface (GUI) application for an automated Online Quiz System. The application provides a user-friendly platform for students to undergo digital assessments and for administrators to manage evaluation data efficiently. Developed using the PyQt5 framework, the system offers a responsive and professional interface, while SQLite is utilized for lightweight, local database management to store user credentials and question banks. The proposed system ensures high accessibility by operating in an offline environment, reducing dependency on constant internet connectivity. Key features include secure user authentication, real-time score processing, and a structured management module for quiz content.***

Keywords: Python, PyQt5, SQLite, GUI, Quiz Management System, Software Development.

I. INTRODUCTION

In the contemporary educational landscape, digital assessment tools have emerged as a viable alternative to traditional paper-based testing methods. These modern systems ensure efficient data distribution and enhance the evaluation process by providing a secure and immediate feedback environment. While web-based platforms are common, they often face limitations in areas with unstable internet infrastructure. The demand for robust, desktop-based educational software has led to the adoption of versatile tools like Python and the PyQt5 library, which allow for the creation of professional cross-platform applications. By integrating a local relational database like SQLite, developers can ensure that data remains protected and accessible without the need for complex server configurations. This approach is particularly useful for institutional environments where data privacy and system reliability are critical. The remainder of this research paper is organized as follows: Section II presents a Literature Survey of current digital assessment technologies. Section III (Principles of the Automated Quiz System): Explains the underlying logic of event-driven programming and the separation of visual design from operational code. Section IV (Components of the System Architecture): Provides a detailed study on the integrated technical elements, including the PyQt5 interface, the SQLite engine, and specific navigational widgets like the Next, Review, and Submit buttons. Section V (System Workflow and Logical Design): Illustrates the step-by-step process of user interaction and data persistence through a comprehensive process flowchart. Section VI (Benefits and Performance Analysis): Discusses the advantages of the system, such as offline reliability and efficient data management. Section VII (Future Prospects) and Section VIII (Conclusion): Outlines potential technological advancements and summarizes the contribution of the study to digital assessment tools.

II. LITERATURE SURVEY

The shift toward digital assessment systems has become a central theme in educational technology research, aiming to solve the inefficiencies of manual grading and delayed feedback. This section reviews recent academic contributions that utilize Python-based frameworks and structured database management to modernize evaluation. Seetha et al. (2023) explored the versatility of Python in creating domain-specific utility tools. In their work, *"A GUI Based Application for PDF Processing Tools Using Python & CustomTkinter,"* [1], they highlighted how Graphical User Interfaces (GUIs) significantly enhance user accessibility for complex technical tasks. While their study focused on document processing, the underlying principle of using Python to bridge the gap between complex backend logic and intuitive frontend design provides a foundational basis for our offline quiz system. A more direct application of digital testing is found in the research by Khare, Kumar, and Singh (2026). In their paper, *"AI Driven - Online Quiz Assessment System,"* [2] published in the *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, they discussed the integration of automation in scoring and student monitoring. Their findings suggest that digital platforms not only reduce administrative overhead but also provide a more secure environment for competitive examinations.

©IJRASET: All Rights are Reserved

However, their model primarily targets online environments, leaving a gap for institutional settings with limited internet infrastructure. Complementing this, Ahmed and Hossain (2020) provided an in-depth analysis of structural requirements for academic software in the *International Journal of Emerging Technologies in Learning (iJET)*. Their study, *"Web-Based Online Assessment System for Educational Institutions,"* [3] emphasized the importance of real-time score processing and secure data storage. They noted that institutional trust in these systems relies heavily on data integrity and the ability to link unique student profiles to historical performance records.Despite these advancements, many existing tools lack the specific "offline-first" reliability required for local environments where constant server connectivity is not guaranteed. Our research aims to bridge this gap by combining the sophisticated interface capabilities of PyQt5 with the zero-configuration, ACID-compliant local data persistence of SQLite.

III. PRINCIPLES OF THE AUTOMATED QUIZ SYSTEM

1) The system operates on the principle of event-driven programming, where user actions (like clicking a "Submit" button) trigger specific Python functions.
2) The Separation of Concerns principle is applied by using Qt Designer for the visual interface (UI) and Python for the operational logic.
3) The Relational Data Principle ensures that each user's unique ID is linked to their specific test results within the SQLite database, preventing data overlap.
4) The system utilizes conditional logic to instantly compare user input against the "Correct Answer" key stored in the database, allowing for real-time grading.

IV. COMPONENTS OF THE SYSTEM ARCHITECTURE

A. *Front-End Interface (PyQt5 Framework)*

The user interface is the primary layer of interaction, designed to be intuitive and responsive.

- Qt Designer: Used to create .ui files, ensuring a professional and consistent layout across multiple windows such as Login, Quiz, and Results .
- PyQt5 Widgets: The system utilizes specific classes like QMainWindow for the primary container, QLineEdit for secure credential entry, and QStackedWidget to manage seamless transitions between different application states.

B. *Back-End Database (SQLite Engine)*

The system utilizes SQLite for structured data persistence, organized into a relational schema that ensures data integrity and efficient retrieval. The database architecture consists of the following primary tables:

- Questions Table: This core component stores the academic content of the quiz. It includes fields for the unique question ID, the question text, multiple-choice options (1, 2, 3, 4), and the correct answer key used for automated grading.
- Results Table: This table acts as a historical log for student performance. It captures the unique result ID, the associated User ID, the total marks obtained, and the specific date or timestamp of the assessment.
- Users Table: A dedicated table for authentication and profile management. It stores user credentials, including usernames and passwords, along with unique identification numbers to link students to their respective result records.

C. *The Connectivity Bridge (sqlite3 Library)*

The interaction between the Python operational logic and the SQLite file is managed through the sqlite3 module.

- Structured Query Execution: The system uses standard SQL commands such as SELECT for question retrieval and INSERT for saving student scores into the Results table.
- Data Mapping: The architecture ensures that when a user interacts with the Next or Submit buttons, the system maps the current user's session ID from the Users table to the new entry in the Results table, maintaining a clear relational link.

V. SYSTEM WORKFLOW AND LOGICAL DESIGN

We have two types of users for this examination system: Teacher and Student.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

A. Operational Workflow: Student Module

Step 1: System Initialization

- The application launches and loads the graphical user interface (GUI).
- A secure connection is established with the backend SQLite database.

Step 2: User Authentication

- The student enters their login credentials (e.g., username and password).
- The system validates these credentials against the Users table in the SQLite database.

Step 3: Test Category Selection

- Upon successful login, the "Category Selection Window" is displayed.
- The student selects their desired difficulty level for the test (e.g., Easy, Medium, or Hard).

Step 4: Examination Interface & Monitoring

- The system loads and displays the test question paper relevant to the selected category.
- The examination timer is simultaneously initiated, and real-time monitoring begins.

Step 5: Automatic Evaluation & Result Storage

- The test session is concluded, triggered either by the timer expiring or a manual submission by the student.
- The system automatically evaluates the responses and calculates the final score.
- The calculated score is displayed on the screen to the student, written back and stored securely in the database.

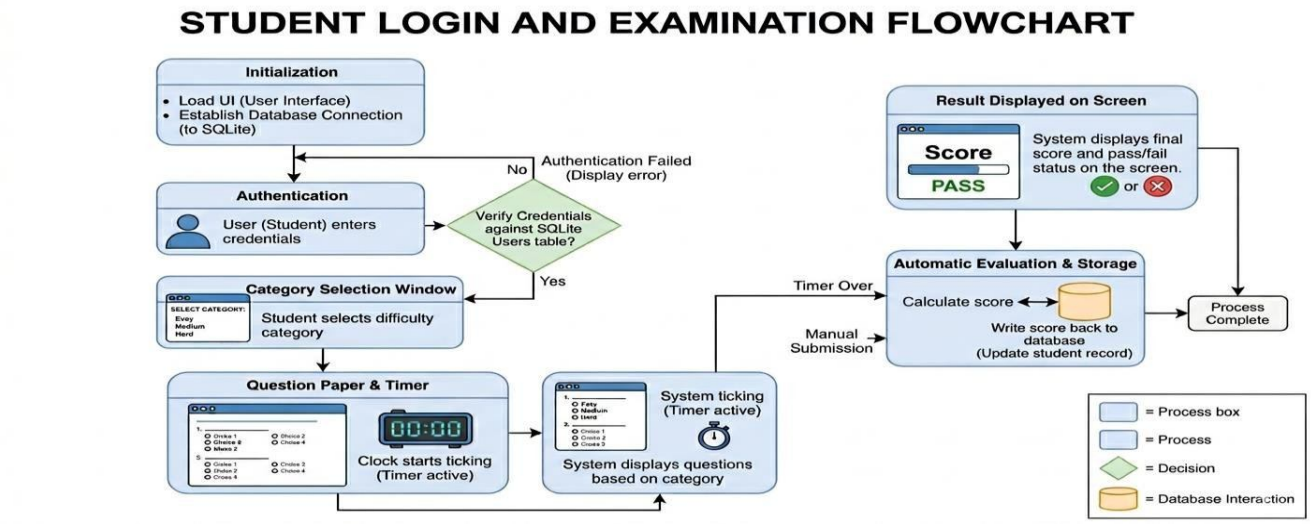


Fig. 1 Student Module Flowchart

B. Operational Workflow: Teacher Module

Step 1: System Initialization

- The application launches and loads the graphical user interface (GUI).
- A secure connection is established with the backend SQLite database.

Step 2: Admin Authentication

- The teacher(admin) enters their login credentials (e.g., username and password).
- The system validates these credentials against the Users table in the SQLite database.

Step 3: Teacher(Admin) Dashboard

- Upon selecting the "Add Question" option, the admin is redirected to a Category Selection Page. Here, the teacher must specify the difficulty level (e.g., Easy, Medium, or Hard). Once the level is selected, the system displays the dedicated interface for authoring and committing the new question to the database.
- Choosing the "View Questions" option provides the teacher with a read-only list of existing questions. This interface is strictly for auditing purposes and allows the teacher to review the full content of previously established questions.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

- When the "Delete Questions" option is selected, the system navigates to a Category Selection Page. The teacher must first choose the difficulty level corresponding to the question they wish to remove. Following selection, a filtered list of questions within that category is presented for secure deletion.
- The "Set Quiz Timer" option initiates a workflow to configure the examination duration. The teacher is first directed to a page where they can set the category and the time. This ensures that different timers can be established for each difficulty level (e.g., Easy exams may have a different duration than Hard exams). After specifying the category, the system enables the teacher to input the required time limit for that specific module.

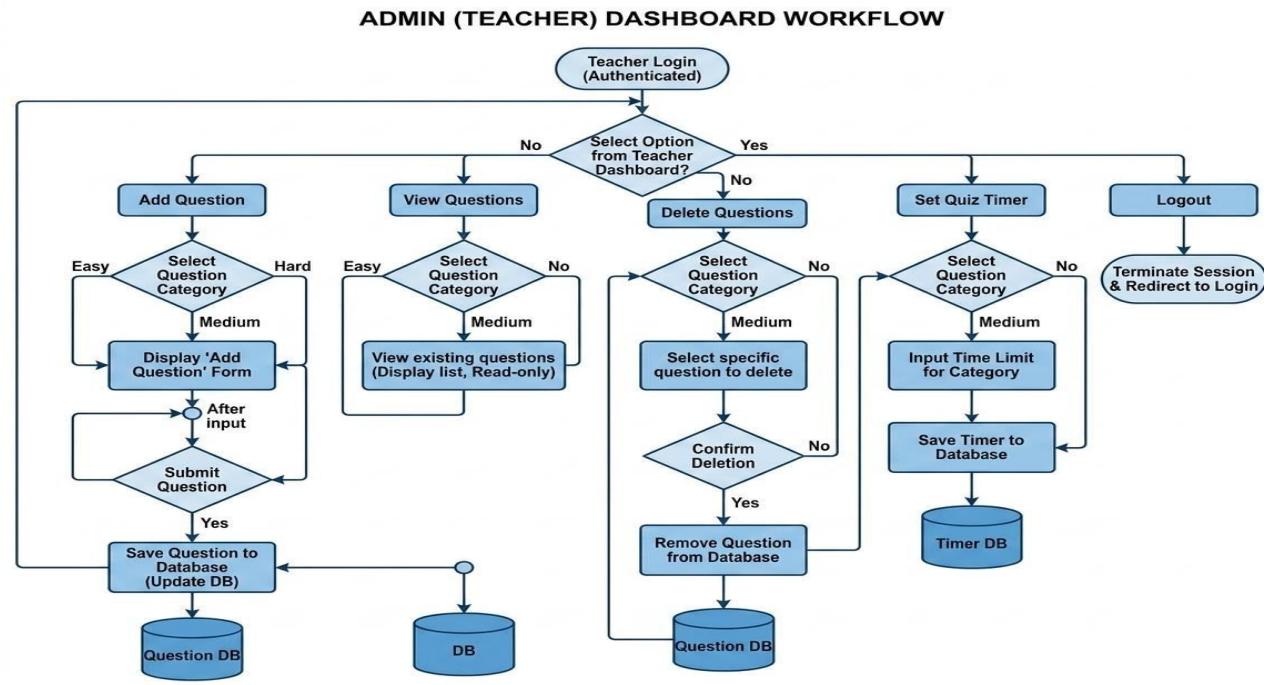


Fig. 2 Teacher Module Flowchart

VI. BENEFITS AND SYSTEM PERFORMANCE

1) Zero-Latency Access: Local database queries provide faster result retrieval than cloud-based systems.
2) Offline Reliability: The system functions perfectly in areas with no internet, ensuring academic continuity.
3) Data Integrity: SQLite's atomic transactions ensure that student progress is not lost even if the application closes unexpectedly.

VII. FUTURE PROSPECTS

The current version of the Offline Quiz System provides a foundation for several advanced technical integrations:

1) AI Proctoring: Implementing computer vision logic to monitor user behaviour via webcam for automated exam invigilation.
2) Cloud Synchronization: Transitioning from a purely local SQLite model to a hybrid system that syncs results with a centralized cloud database.
3) Automated Question Generation: Integrating Natural Language Processing (NLP) to generate quiz content automatically from uploaded study materials.
4) Cross-Platform Portability: Adapting the existing PyQt5 logic for mobile deployment on Android and iOS devices.

VIII. CONCLUSION

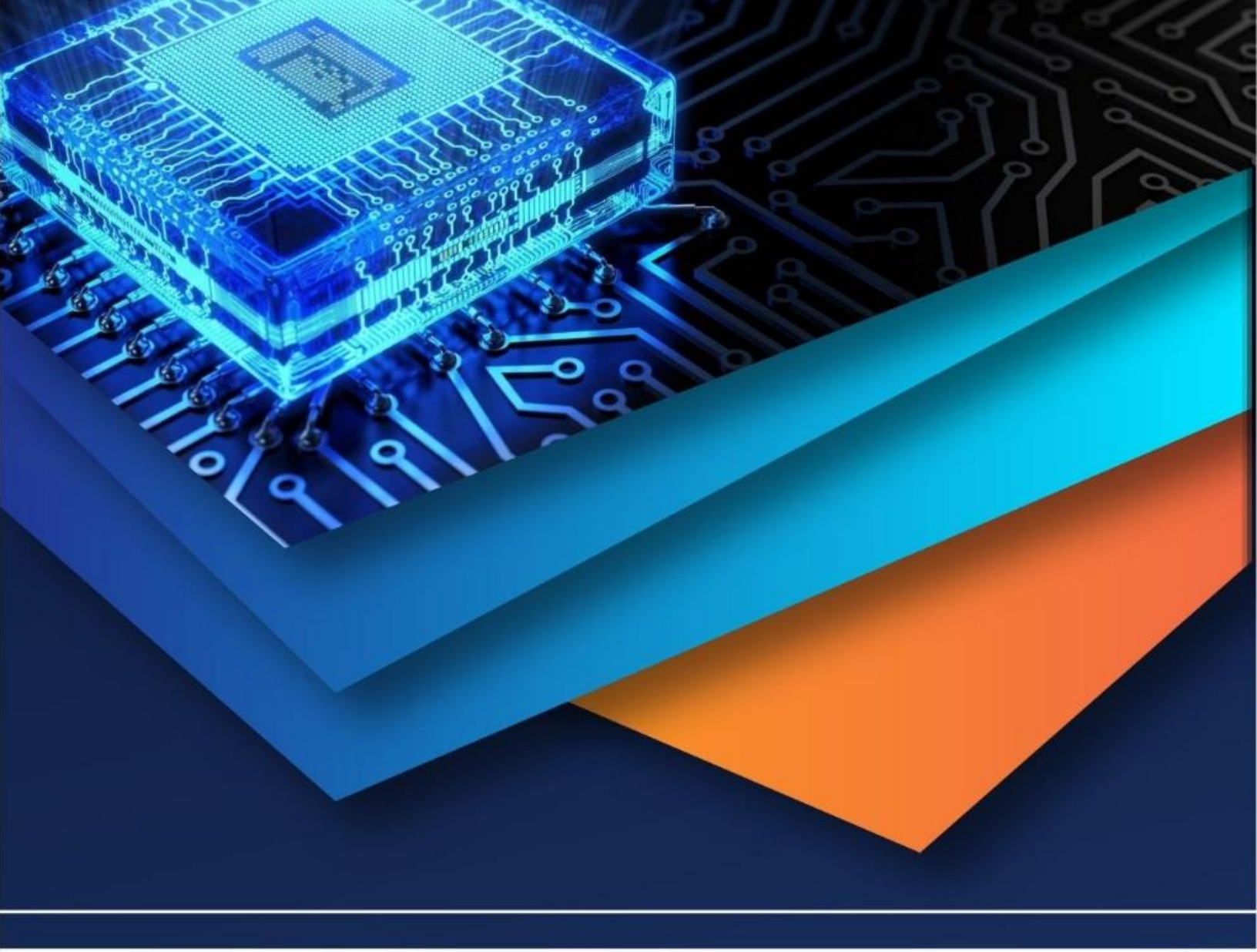
This research successfully demonstrates the implementation of a high-performance, offline-capable digital assessment tool using Python, PyQt5, and SQLite. By utilizing an event-driven GUI and a relational database schema, the system ensures 100% availability in environments with limited internet connectivity.

The study highlights that desktop-based applications, through structured navigation (Next, Review, and Submit), can provide a professional and secure examination experience. Ultimately, this project offers a sustainable and efficient alternative for educational institutions to modernize their evaluation processes with minimal infrastructure costs.

REFERENCES

[1] Seetha, H., et al. "A GUI Based Application for PDF Processing Tools Using Python & CustomTkinter." International Journal for Research in Applied Science & Engineering Technology (IJRASET), vol. 11, no. 1, Jan 2023.
[2] S. Khare, A. Kumar, and R. Singh, "AI Driven - Online Quiz Assessment System," International Journal for Research in Applied Science & Engineering Technology (IJRASET), vol. 14, no. 4, pp. 3854–3858, Apr. 2026.
[3] M. Ahmed and S. Hossain, "Web-Based Online Assessment System for Educational Institutions," International Journal of Emerging Technologies in Learning (IJET), vol. 15, no. 12, pp. 110–122, Dec. 2020.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)