Improved and Simplified Online testing,

Aleksandre Kobaidze, Nikoloz Robakidze

e-mail: aleksandre.kobaidze@tsu.ge,
nikoloz.robakidze383@ens.tsu.edu.ge

Computer Science Department,Faculty of Exact and Natural Sciences, Tbilisi State University, 1 Ilia Tchavtchacadze Avenue

Academic institutions have faced increasing challenges due to the widespread adoption of artificial intelligence (AI) tools in everyday academic activities. The rapid proliferation of these technologies has complicated the assessment process for essays, homework, and other assignments. Compounding this issue is the unreliability of many so-called "AI detection tools," which are known for generating false positives, further complicating the responsibilities of lecturers and instructors. This project proposes a solution aimed at providing educators with an accessible, reliable, and easy-to-use platform for conducting quizzes and assessments. Unlike some commercial testing platforms—such as the Pearson testing suite, which offers robust security but lacks flexibility—our solution is designed to adapt to the diverse assessment needs that arise over the course of an academic semester. Meanwhile, simpler platforms such as Moodle, though widely used, suffer from configuration complexity, limited scalability, and performance issues during high-load scenarios. Furthermore, Moodle's general-purpose design results in an overloaded interface with numerous unused plugins and features, which can hinder usability. Our proposed system aims to strike a balance between security, usability, and adaptability. The platform is lightweight, resistant to cheating, and straightforward to configure. It leverages a set of established technologies to ensure stability, reliability, and ease of deployment. Instead of relying on a centralized server infrastructure, the system allows instructors to host individual testing sessions locally. The use of Google Forms as a backend ensures fault tolerance and minimizes resource requirements. A custom Domain Name System (DNS) configuration—implemented via Pi-hole—restricts access to AI-powered services such as ChatGPT, Claude, Gemini, and others during assessments. Additionally, client-side scripts enforce fullscreen mode and restrict navigation outside the test environment, thereby minimizing the opportunity for academic dishonesty.

Technologies Used:

- **HTML, CSS, and JavaScript:** Used for window control, timers, and user interface elements.
- **Google Apps Script:** Facilitates integration with Google Forms for data collection and test management.
- **Pi-hole:** Provides DNS-level filtering to block access to prohibited websites during assessments.
- **Docker:** Ensures portability, ease of deployment, and future scalability of the platform.
- **Bash Scripting:** Automates the system startup process for streamlined use in test environments.