EDUCATIONAL SECURITY AND DIGITAL CHALLENGES

Authors:
Abhay Verma
Dhariya Rathi
Manan Verma
Akshit Pundir
Devyansh Rajput

The Problem

In today's time security regarding education has arised as an issue. As education has become prevalent on the internet countless digit challenges are faced by students and educators alike.

Problems such as cheating, unethical use of AI and compromising of personal data and data breaches have come-up and hinder the process of education and learning along with being highly damaging to one's personal life and mental well being.
It is very easy for students to cheat on online quizzes and tests, so we have decided to find a solution for this complication.

The Solution

We have designed and programmed the project so as to reduce the chances of the student being able to cheat and deceitfully get a high score. We use a basic Social Science quiz as the test and have implemented features to achieve authentic test scores.

1. **Tab Protection:**
   This feature ensures that the tab that's opened on the system, the quiz tab, stays open. What that means is that the student only stays on the quiz and does not open any other tab so as to cheat. If the students were to open any-other tab, he/she would be kicked-out of the quiz.
2. **No Right-Click:**
   Many students right-click to inspect the web-page and enter the database through which they write the right answers. This sort of cheating even went viral at a time of COVID. The solution is simple. Right-clicking on the quiz would is disabled and nothing would happen if it were to be done.
3. **No Re-entering:**
   Some students try to take a quiz with different users, so that they can see the questions. This is also solved by adding a sign-up system which makes sure that the same user does not try to take the quiz multiple times. An existing user can only take the quiz once, so he/she cannot do it twice and earn a higher score.

4. **Face-Detection:**

   Some students try to take the test by using unethical methods to get answers of questions in their physical place of test-taking, this allows them to get a higher score and the quiz would not be of the utmost use and effectiveness.

   To solve this issue a face detector which detects the head movements of the test taker is programmed into the test page.The camera is used to ensure that the student remains focused solely on the screen and does not engage in any other activities, thereby reducing the risk of cheating. If it detects that the user’s head has moved towards the side and not towards the screen, it will show a warning to the user, instructing them to face towards the screen.

Details

The project is a quiz giving Website with the aforementioned features to detect and restrict cheating. It is mainly divided into three parts:

Front-End:
The front-end is the face of the website and is programmed mainly with the use of HTML. Apart from that CSS and JavaScript are also used in places to make the website more professional and user friendly.

Back-End:
The back-end is the hidden logic of the website like score calculator and the features to respond to cheating. It is mainly coded with the help of Python.

Integration:
The connection of the back-end with the front-end is essential for any website to work efficiently. We have used a Python library known as Flask to achieve success in this web-development project for educational security and to overcome its challenges. Flask made it simple and efficient to allow for easy connection between the two faces and the process flow smoothly so that the user does not feel any discrepancies in the use of the website.

FUNCTIONING

The main functioning of the project is in its integration. First of all, there are many .html files for the various web pages like test_questions, sample_questions and maintest. CSS is used to enhance the GUI and make the page user friendly.

Then, Python files are used for back-end programming and camera feature. The scoring system, questions, random order for set questions and the redirection to various pages as per call from front-end is done in the Python files.

Finally, JavaScript is used for calling the back-end and to give a time-limit for the quiz (1 hour), it is also minorly used in other places to make the website run smoothly.

Once a button is clicked, for example the “Take Quiz”, then using JavaScript a call is given to back-end python which then gives a response to redirect the user to the Quiz Web-Page with all the anti-cheat features enabled.