

Python Application for Conducting Online Examination

Mohammed Nihal Shareef, Prabhuthvam Chakali, Abhishek Ghogre and Reddy Saisindhuteja

Department of Computer Science and Engineering, Vardhaman College of Engineering, Hyderabad, Telangana, India

Correspondence to:

Mohammed Nihal Shareef
Department of Computer Science and Engineering,
Vardhaman College of Engineering,
Hyderabad, Telangana, India.
E-mail: mohammednihal782@gmail.com

Received: September 19, 2023

Accepted: December 05, 2023

Published: December 08, 2023

Citation: Shareef MN, Chakali P, Ghogre A, Saisindhuteja R. 2023. Python Application for Conducting Online Examination. *NanoWorld J* 9(S4): S514-S517.

Copyright: © 2023 Shareef et al. This is an Open Access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CCBY) (<http://creativecommons.org/licenses/by/4.0/>) which permits commercial use, including reproduction, adaptation, and distribution of the article provided the original author and source are credited.

Published by United Scientific Group

Abstract

Nowadays exams are all conducted online, and the marks allocation is being automated. The proctoring of the exam session is done in various ways based on the importance of the exam. The certification exams, hiring/recruitment exams are hosted in online mode and assessment is also automated. There will be no requirement of human interaction for monitoring the assessment and validation of the answers provided. But the exams which are being conducted in an engineering college for practical assessment are not online and require human interaction for monitoring, validating, and assessing the solution. To reduce the human effort from the side of the examiner and the examinee, a platform is proposed, using which the many practical examinations of engineering colleges related to computer science and information technology can be hosted.

Keywords

Python, Online exam, Monitoring

Introduction

Engineering students have at least 2-3 courses per semester which are integrated with the laboratory tasks. In these courses separate exams will be conducted for theory and practical, both have their weightage and are important to score good in that course. Lab examinations are conducted for a course in a semester 2-3 times (internal and externals). The examiner combines 2-3 questions and makes it into a set. A total of 4-5 sets will be made. Each student will be given a set such that no two adjacent students will get the same set. Every student will be given a booklet in which the solution to the questions should be written. The written solution will be checked by the invigilator, who then decides whether the student can execute the code or not. After getting validated, the student will open the PC and start writing code in any online compiler or the editor or any IDE based on the course. After writing code and checking output then invigilator then checks the code written and validates the output of it. Marks will be awarded to the student based on the presentation of the solution in the booklet, and the execution of the solution. The entire process is time taking and requires a lot of human effort. The generating of a set of question papers, and distribution of it to students will be automated and can be done by the server itself. The marks allocation for each student and validating of the solution is automated. Partial marking for the solution will be given not only based on the performance, but the similarity of the solution to the actual solution [1, 2].

Not only will the conduction of the exam be done online, but the flaws like student cheating, taking information from the material in the pc or browsing on the internet, these all will be handled. As the examination starts, the student cannot switch to another window and cannot use any shortcuts, few keystrokes will not be responding until the completion of the test [3].

The scheduling of the exam can be done by faculty by registering to the web-

site, the examiner will have to login and register a course by giving the course name, course code, start time and end time of the exam and the list of eligible students. After registering for a course, the examiner should upload the questions and their details like problem statement, constraints, input format, output format, input test cases, output for each test case and the solution [4, 5].

When the time of the examination starts only the eligible students will log in to the website with their roll number and the course code, then each student will be given 2 questions, which they should complete before the time completion. The marks will be given based on the number of test cases passed and the similarity of the code to the actual solution. While the concept of conducting examinations online using Python predominantly revolves around software and programming, the integration of nanotechnology can enhance certain aspects of the overall system. Python plays a significant role in nanotechnology by providing tools and frameworks for simulations, quantum computing and educational applications. Its versatility and ease of use make it a valuable asset in the multidisciplinary field of nanotechnology [6-8]. Nanosized sensors embedded within examination devices, such as webcams or input devices, can play a role in enhancing security measures. These sensors could capture and analyze biometric data, providing an additional layer of identity verification for online exams.

Experimentation

Existing system

There are some platforms existing where hackathons and events can be hosted. Hackerrank, Codechef can be used to host custom events. But the assessment won't be strictly proctored. Keyboard shortcuts will be enabled, the user can switch between applications or even browse for information. The questions given to the users will not be randomized and everyone will be getting the same questions. Whereas in Codechef every user has a rating, based on the rating the questions will be given according to the difficulty level. Both of the platforms are useful for hosting competitive programming contests. The contest will be open only to registered students, but the proctoring isn't good. The score on each problem is given on the test cases passed. In some cases, only if all the test cases are passed, then the full score is given, otherwise the score on that problem is zero [9, 10].

Mettl is a platform where many types of exams can be hosted. Any certification exams or recruitment exams can be hosted here. The platform can be used to customize the proctoring, i.e., screen sharing, proctor via camera, if we want keyboard shortcuts can be disabled. It takes the microphone input into consideration, microphone and camera are needed, and it has its separate infrastructure readiness check for a system to check whether it is compatible for taking the exam. When the exam is started, a video will be recorded for monitoring the candidate (in some cases random photos will be captured to check student's activity) and a screen will be shared, to monitor the screen for any movement. All these activities run while taking the exam. Some systems may not be compatible for this assessment. The background apps are not checked and closed

when using this platform such that any process running in the background may alter the assessment. Well in the proposed system, all the processes that run in the background will be monitored and any unnecessary processes which are not being depended on by the browser will be killed. Any application which may alter the assessment will be closed. The requirement of the system is browser compatibility, if a browser can run on the system and just the usage of the browser is enough as the infrastructure check for the assessment. Any camera, Microphone, new application are not needed for taking the assessment [11].

Proposed system

The application has 2 different clusters of processing, where one cluster does the processing for examinee in hosting the exam and the other cluster is responsible for the processing of student's requests while attempting the exam and handling the virtual run time environment on the software.

Host exam

The examiner will have to register to the application via the website by giving their name, email, and a password. The email and the password will be used for logging in. The examiner, after logging in will be shown 2 options i.e., host exam and upload question. In the host exam section, the details of the course, timing of exam and the list of eligible students should be entered. After hosting the exam, the questions which should be given to students should be uploaded, for each question, problem statement, question name, input and output format, input and output test cases with the solution should be submitted. Whenever a question is uploaded all the test cases will be executed with the given solution to check the functionality and only if all satisfy the question is stored. Whenever an exam is hosted on a particular course, the same questions can be used in the next examination. Also, there is no limit that the questions uploaded by the user should be used in only a single exam (Figure 1).

Attempting exam

Software will be available on the website which should be downloaded by the students or the examinee for attempting the exam. The user should enter details like roll number

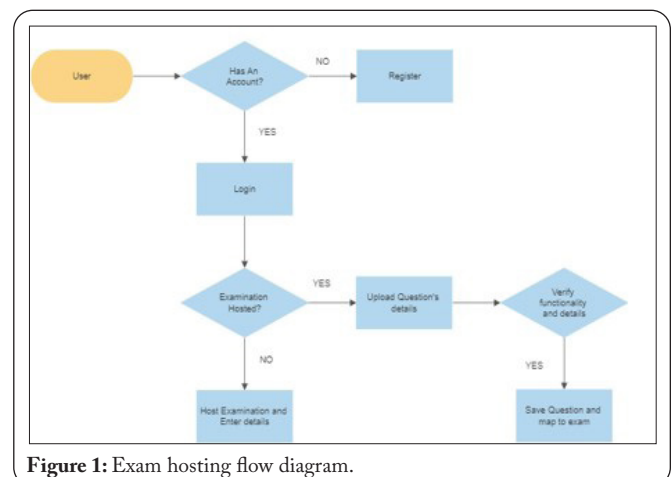


Figure 1: Exam hosting flow diagram.

and the course code. If the given roll number is in the list of eligible students and the exam has started then only the questions panel will be opened, else an appropriate message will be shown (Figure 2). If the user details are correct, then 3 processes will start whose effect will last until the test is submitted, they are:

Blocking shortcut keys

Any shortcut keys like Ctrl+A, Ctrl+S, Ctrl+C, Alt+F4, etc., will all be blocked. The keyboard will not be responding to any commands which involve the keys Ctrl, Alt, Win. Those keys will be blocked until the test is submitted.

Background process verification

The processes which are running in the background, the applications which are open in background which have a chance of altering the outcome of the examination will shut down immediately. Any process or application which is not required and is not dependent for the functioning of the operating system (OS) will be killed.

Full screen mode

The user interface of the software occupies the entire display of the monitor/screen, The examination will be running on the full screen mode, so that the student will not be able to change application or use any files on the system or use any facilities of the computer.

After the above 3 processes are executed, then questions are shown to the students. Each student will be given access to a run-time environment and the questions can be changed in the middle as per their interest. After completion time or student finishing the exam, the marking is done for the code based on the correctness of the code and the marks are stored with the code in the database, so that if the examiner would like to cross verify any of the solutions.

Once the question panel is shown, each user will be given a virtual run time environment.

Results and Discussion

For hosting the exam, the user will have to visit the website and register first, if he does not have any account then log in using their credentials, after logging in, they will be displayed as in figure 3 i.e., name, email, courses registered and 2 buttons for registering courses and uploading questions will be shown. Upon clicking on the course register button, a form will be displayed as in figure 4, where the details of the course, start and end timing of the exam and list of all eligible students should be uploaded, upon successful registering the user will be redirected to the faculty main page as in figure 3. Upon clicking upload question button a form will be displayed, where the details of the questions should be entered, i.e., problem statement, input and output format, constraints, input and output test cases and solution, on successful upload of question the user will be redirected to the faculty main page figure 3.

When it comes to attempting the exam the student will

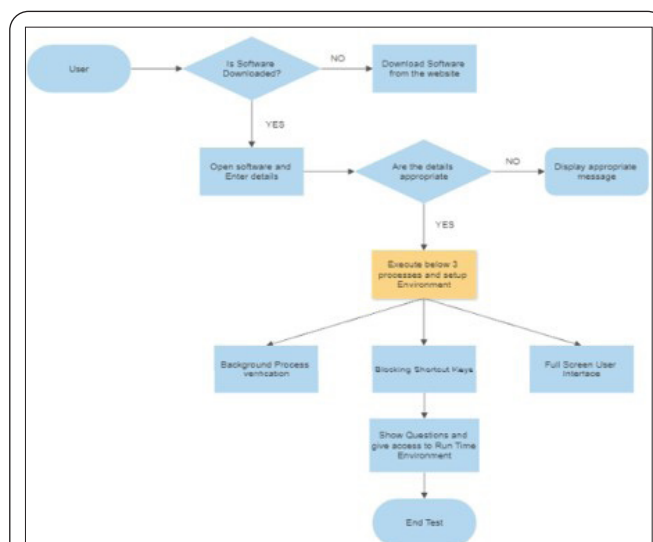


Figure 2: Exam attempting flow diagram.

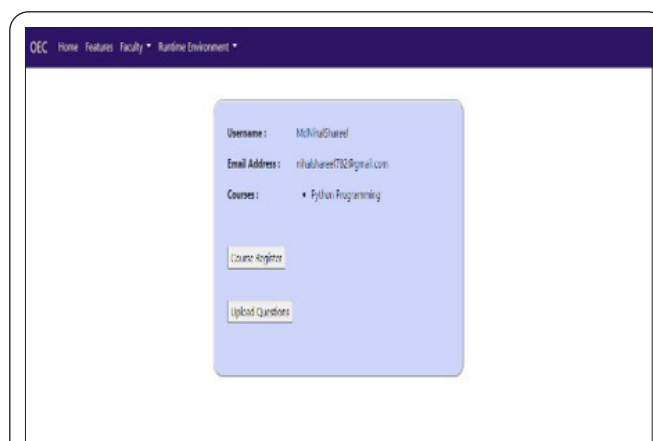


Figure 3: Examinee logged in.

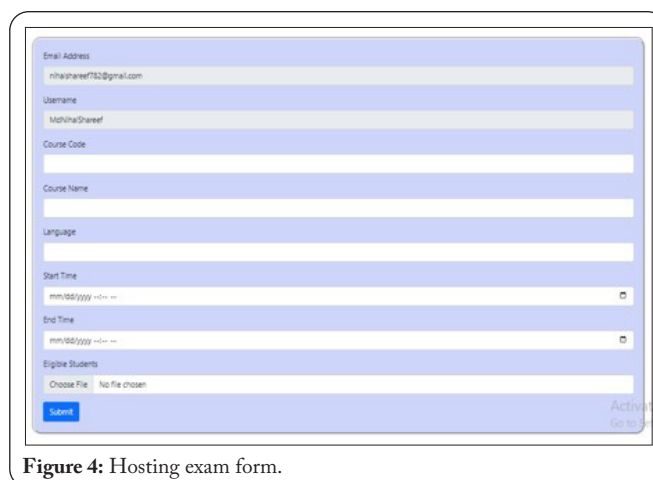


Figure 4: Hosting exam form.

have to download software from the website, upon running the software it will take control of entire screen and 2 input fields will be shown, when the details are entered if the examination is started off given course and the student roll number is the list of eligible students then the user interface will be updated as shown in figure 5, the questions can be switched by clicking on the desired question button and the code can be run by giving any input (Run time environment), else if the



Figure 5: Examination user interface.

details are inappropriate an appropriate message will be shown in the message box while logging in.

Conclusion and Future Scope

In this paper, a platform was introduced which will be used to conduct practical examinations of CS/IT students. This platform itself will handle the distribution of questions to the students, the exam will be proctored effectively such that no application can be used to share the examination info to the other individuals, any shortcuts or switching of applications and tabs in the browser will not be allowed. This application will automate the marks allocation of the students which will now no longer require human interaction for this task. The system requirements for this system to run the application are also not much, just browser compatibility is needed. The reusability of questions will be easier. The same questions can be used for the next or upcoming hosting of exam on the same course.

This system can be extended in a manner such that it can open any applications inside a window, i.e., act like a layer between the OS and the application or software to build any projects. The exam software should act as a virtualization layer between the external software and the OS. The external software can be an application which runs on different OS. For example, the host OS is Windows, and the exam is about OS, which has creation of pipes, shared memory, etc., for that a terminal is needed (of Linux - is preferable). So, the running of a Linux terminal on the Windows host OS is to be done here. In this scenario, there will be 2 layers of virtualization. It can be on 2 types, 1st one is where Linux OS will be running on the exam software and on the exam, software will be running on the host OS (i.e., Windows) or Linux should be running as 1st layer on host OS and exam software will be running on 2nd layer i.e., on Linux OS.

Using the above case any software like VS Code, Eclipse IDE, Android Studio, any tools like Star Uml can be used, but for each software and each question the solution should be uploaded as the compressed zip file and for comparison for each case a separate tool must be constructed.

Acknowledgements

None.

Conflict of Interest

None.

References

1. Kotwal DV, Bhadke SR, Gunjal AS, Biswas P. 2016. Online examination system. *Int Res J Eng Technol* 3(1): 115-117.
2. Abass OA, Olajide SA, Samuel BO. 2017. Development of web-based examination system using open source programming model. *Turkish Online J Distance Educ* 18(2): 30-42. <https://doi.org/10.17718/tojde.306555>
3. Zinovieva IS, Artemchuk VO, Iatsyshyn AV, Popov OO, Kovach VO, et al. 2021. The use of online coding platforms as additional distance tools in programming education. *J Phys Conf Ser* 1840(1): 012029. <https://doi.org/10.1088/1742-6596/1840/1/012029>
4. Böhmer C, Feldmann N, Ibsen M. 2018. E-exams in engineering education—online testing of engineering competencies: experiences and lessons learned. In *IEEE Global Engineering Education Conference*, Santa Cruz de Tenerife, Spain.
5. Sukadarmika G, Hartati RS, Sastra NP, Wiharta DM, Setiawan MA. 2016. Proposed model for e-exam availability in WLAN environment. In *International Conference on Smart Green Technology in Electrical and Information Systems*, Denpasar, Indonesia.
6. Abd Rahim TNT, Abd Aziz Z, Ab Rauf RH, Shamsudin N. 2017. Automated exam question generator using genetic algorithm. In *IEEE Conference on e-Learning, e-Management and e-Services*, Miri, Malaysia.
7. Hameed MR, Abdullatif FA. 2017. Online examination system. *Int Adv Res J Sci Eng Technol* 4(3): 106-110.
8. Choubey A, Kumar A, Behra AR, Kisku AR, Rabidas A, et al. 2020. A study on web based online examination system. In *International Conference on Recent Trends in Artificial Intelligence, IOT, Smart Cities & Applications*, Jharkhand, India.
9. Chang L, Ming XG. 2012. Application research of WEB examination system based on college. *Energy Procedia* 17: 528-533. <https://doi.org/10.1016/j.egypro.2012.02.131>
10. Cartwright R, Steele Jr GL. 1998. Compatible genericity with runtime types for the Java programming language. *ACM SIGPLAN Notices* 33(10): 201-215. <https://doi.org/10.1145/286942.286958>
11. Hang B. 2011. The design and implementation of on-line examination system. In *International Symposium on Computer Science and Society*, Kota Kinabalu, Malaysia.