A Senior Research – An Integrated and Automated Submission System

Kerules Fareg and Ching-yu Huang
School of Computer Science, Kean University, Union, New Jersey, USA

Abstract – *This paper will present a senior project that integrates many technologies to create an Integrated and Automated Submission System (IASS). The system presented in this paper is an online system that can help students and instructors on the path of learning to effectively learn and manage the programming assignments assigned by instructors to the students in such classes. The IASS system presented in this paper provides feedback to students upon submitting the programming assignments to the system by running the programs and alerting the students to any errors occurred or prompting for success. This immediate feedback will help students improve their learning and encourage their success. The system will also help the instructors in managing the workload of the assignments assigned to students by executing and running the programs for them and provide appropriate comments that instructors can use to grade the students. In short, this system can revolutionize the way students learn today.*

Keywords: Database, GUI, submission, automation, Java, Compiling System

1. Introduction

In all the universities and colleges in the United States, students in the computer science and technology fields are required, in their learning tracks, to take programming assignments in order to practice and apply what they have learned in productive, real world examples. Such assignments, though it is very helpful for the students, it can be a huge burden on the instructors that have to compile and run the programs of each student in order to grade it. This might cause delays in the process that results in students not getting the proper feedback in a timely manner.

As a result of this issue, a rise of on-going research on developing an automated submission systems have been going on. Although the work of such research has been tremendous to say the least, most of these systems have been more of dedicated systems rather than general systems of submissions. In this paper, we present the idea of a general submission system that can be used by all teachers and all students who wish to use the system. The IASS system will integrate a Linux system, a web server, and a database. As seen in previous submission systems of such type, the IASS will allow students to login, submit their programs and receive immediate feedback after automatically compiling and running the programs. Teachers will also be allowed to login to the system, add their classes with their respective roster list, and assign the assignments for the students based on their course loads. In addition, the system will check the students' programs for similarities and take action in order to prevent plagiarism.

This generally-integrated system can benefit many schools and teachers in the future by choosing to adopt the IASS system as their platform of assignment submissions. The IASS can be a breakthrough in future of programming education and assignment submissions. As a senior student participating in such research, the opportunity was provided to learn and explore new knowledge far beyond the typical course curriculum. It is helpful to explore the integrated environment between the Linux system and the web system and learning how to execute programs on Linux through a web server. This project also provided the opportunity to learn how to accomplish many tasks in developing and maintaining the system though different platforms and different phases. Such knowledge and research goes beyond the scope of the classroom.

2. Related Work

Earlier Research on this topic, such as the research done by Huang [1] on integrated Automatic Compiling System for Student Feedback on Java Programs took on the same challenge of creating such system and developed a system that students can use to submit their programs and receive their feedback immediately from the system after the programs are compiled and run. Even though, the research was a prototype for such area, the system developed was intended to be a dedicated system to specific classes only.

Huang's research was taken as a reference to this project along with the knowledge gained from the CS course took at school. Such background experience helped tremendously during the development of this system by applying the knowledge learned in the database classes in developing a good database design for this system as well as applying the knowledge learned in my information systems development classes all throughout the phases of development.

3. The Designs

The designed architecture presented here consists of four major components (Figure 1) 1. Login, 2. Submission, 3. Compiling and Execution, 4. Review (only for Instructor's Interface). These four components will serve for the student

interface, the instructor's interface, and the administrator's interface. The Files submitted to the system by the students will be saved in their respective folders on the Linux server under the instructors' accounts.

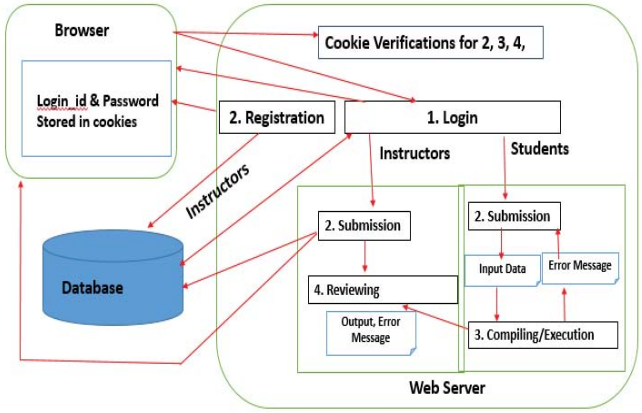

FIGURE 1.
THE SYSTEM ARCHITECTURE DIAGRAM

The Login Component includes 3 different roles as shown in Figure 2. Each user role has his/her own interface according to his/her access rights and the intended use of the system.

FIGURE 2.
THREE MAIN ROLES OF THE SYSTEM

3.1 Student Interface Architecture

3.1.1 Login Component

All students are required to login to the student portal (Figure 3) before performing any activities. Students will not be required to register to the system because the system will pre-register them when the instructor uploads the roster file to the system. After a successful login, the student user will be shown a page summary of his/her current grades for all the classes he/she has registered with the system. In addition, the page will also display the upcoming assignments for all the student's classes with their description and due dates. Students can then navigate the form to upload their files and submit the assignments right from the home page.

FIGURE 3.
STUDENT LOGIN PAGE. INSTRUCTOR AND ADMINSTRATOR HAS A SMILIAR PAGE.

3.1.2 Submission Component

This component will be used by the students to submit their assignments. All files submitted to the system will be checked for validity to insure that students submit the right type of files. The component will also be done using cookies for security purposes and to insure that the person who is logged in is the only person who can view and submit the assignments. The files submitted with the submission component will first be uploaded to the Linux server under the class instructor's account and then compiled and executed thereafter according to the description of their assignments.

3.1.3 Compiling and Execution Component

After the program file is submitted and uploaded to the server, the system will then utilize the Linux standard output 1 and standard error 2 and use it to invoke the Linux command system to compile and run the program with the given file name. Finally, the program will output all the compiling errors or the runtime errors with the program, if there are any, or output the results of the program to the screen with a success message. When the compilation and execution components are done, the results will be saved/changed accordingly in the instructor's reviewing component.

3.2 Instructor Interface Architecture

3.2.1 Login Component

Same as per students, all instructors have to login to the system in order to use it. Instructors who do not have login credentials will be instructed to signup first before using the system. All the authentication processes done by the system are secure and safe. Upon successful login, the instructor will be shown a summary page where all students' grades will be displayed according to their respective rosters along with links to the different components of the system for instructors such as the submission component and the reviewing component.

3.2.2 Submission Component

The submission component for instructors is a little different than it is for the students. Instructors will use the submission component of the system to upload and submit a roster file for each class the instructor wishes to add to the system and create assignments and submit them to students.

3.2.3 Reviewing Component

After a successful login by the instructor, the user will be redirected to a home page with a button to the reviewing component in the system. In the reviewing component, instructors will be able to view all the programs submitted by the students according to their classes and their assignments. The instructor will be able to choose which class he/she wants to view the assignments for and which assignment he/she would like to view the submitted programs for. The submitted programs in the reviewing component will display according to the students' names and will also display all the errors that occurred when running/compiling the program on the student interface, if any, or it will display the results of the program if it ran successfully with the timestamp of the last submission.

If a student did not submit the source code for the assignment, a red-colored message saying "Missing Source Code" will be displayed for the instructor to see under the student's name. The system will also display a message to the instructor promoting him/her if there were any plagiarism detected by the system for any submission by the student for the selected assignment, along with sending the instructor an email with set message when the plagiarism is first detected by the machine during the compiling and execution component on the student interface. Instructors can then review the submitted program with the output and give the students the grades accordingly.

3.3 Administrator Interface Architecture

The administrators are subjected to login to the system as well. The interface for the admin role is different, however, because of the jobs that they do with the system maintenance. The administrator interface will include an interface of all the tables in the database and the teachers' information associated with it. The system will also allow the administrator the drop function of the tables at any time. This function will be used by the admins to maintain the system after each semester by getting rid of the tables that are not in use by the instructors.

4. Technology

The IASS system was developed using multiple technologies integrated together. In terms of PHP development, the PHP 5 manual and documentation [5] were referenced multiple times during the development of this system along with the PHP 5 tutorial provided by w3school [2]. In addition, the MySQL database documentations and tutorials provided by W3school [3] was referenced during the design and the implementations of the database. In addition, the shell_exec command [4] was also used which provided the ability to execute Linux commands via shell and provide the output in PHP. The system is divided into three components as shown below:

4.1 Graphic User Interface

The frontend Graphical user interface was developed using HTML and PHP with graphics and animations using Bootstrap and CSS. The interface is used to communicate with the user through the web browser.

4.2 Application Server

The application server was developed using PHP to communicate with the frontend and the backend in a secure manner. The application server consists of commands to the database and the Linux server based on the user's request. All database requests are developed in a secure manner using PHP.

4.3 Database server

The database for this system consists of a MySQL database containing all the tables needed for login and registration purposes. In addition, there are tables designed for the courses according to each course added by the instructors and also tables for the assignments assigned by the instructors to their rosters.

5. Conclusions

The IASS system presented in this paper can have a major impact on the way students learn today and in the future. This system can revolutionize the way instructors give out assignments along with the way students learn and practice programming with the feedback given when students submit their programs. The system can also help instructors a great deal with managing the programming assignments in their classes. The system presented in this paper takes minimal efforts for instructors to review and grade the assignments along with helping the students get the immediate feedback on their submissions to enable them to make the necessary changes if possible in a timely manner.

Developing this system helped a lot in learning and exploring some new technologies that are not taught in classes before as well as applying the technologies learned previously on a well thought project such as this one. More importantly, this senior research provided a good opportunity to learn how all these technologies can be integrated and worked together.

This system can be expanded beyond the scope of the current system to incorporate new functions. Such functions might include an automatic grading function for the assignments submitted by the students that will produce the

grades for the students and calculates it based on the grading rules defined by the instructor. Another function that can be added to the system in any future work is a function that can generate tests or assignments for the students based on the given criteria and a given set of questions the system can choose from and generate the tests. Lastly, future functions for the administrator can be added such as maintaining a list of teachers' and students' information and the ability to manually register teachers if need be. However, these functions are beyond the scope of this paper and can be designed and incorporated to the system in the future.

6. References

[1] Ching-yu Huang and Patricia A Morreale, *An Integrated Automatic Compiling System for Student Feedback on Java Programs*, Proceedings of 2015 IEEE Integrated STEM Education Conference (ISEC), Princeton, New Jersey, March 7, 2015.

[2] PHP 5 Tutorial, http://www.w3schools.com/php/

[3] MySQL Database, http://www.w3schools.com/php/php_mysql_intro.asp

[4] shell_exec — Execute command via shell and return the complete output as a string, http://php.net/manual/en/function.shell-exec.php.

[5] PHP 5 Manual and Documentation – Reference Documentation for PHP programing, http://php.net/manual/en/intro.spl.php