iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** XI **Month of publication:** November 2022

DOI: https://doi.org/10.22214/ijraset.2022.47495

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 10 Issue XI Nov 2022- Available at www.ijraset.com

Synergic Coding System: An Online Coding Platform

Madhusudan Sanjay Patil[1], Shubham Nandkishor Deore[2], Mohit Hitendra Bisht[3]
[1]*Department of Information Technology KBT College of Engineering, Nashik*
[2, 3]*Department of Computer Engineering Terna Engineering College Nerul West, Navi Mumbai*

Abstract: Online education is presently quite popular. The widespread availability of the internet makes information sharing and acquisition simple and affordable. Many schools now offer online courses in all subject areas. More students are turning to online learning due to its simplicity and ease. Online coding boot camps grew by 177% between 2018 and 2019, according to a private organization that assists consumers in finding the most affordable coding boot camps, based on a market analysis of 79 US and Canadian locations.

With the increase in online coding classes and boot camps, there is a more significant demand for tools to assist instructors and students in reaping the valuable benefits from the new methods of learning and giving a lecture. The primary issue with this type of learning is the environment in the coding or programming sector. As building up infrastructure and an environment is critical. Another concern is the interaction between the teacher and student because progress tracking is unavailable in the existing systems.

In this research, we put light on providing the possible solutions for two prominent issues:

1) Environment Configuration
2) Progress tracking

This report specifically focuses on understanding the existing systems and incorporating these to develop unique software of our own.

Keywords: online education, coding boot camps, interaction, environment configuration, progress tracking

I. INTRODUCTION

Online classrooms have become increasingly prevalent across all academic disciplines. Gradually, learners and educators. Unlike the olden days, the internet has become highly affordable. It is readily accessible to every nook and corner of the world, which makes it uncomplicated and cost-effective to disseminate and attain knowledge. These days academies increasingly offer online courses, particularly in the computer science field. It comprises a variety of study domains such as web development, android development, machine learning, etc.

After the digital revolution, every other sector and industry is related to computer science. Consequently, there is a high demand for individuals who can code and understand computer science concepts. Also, most employment is available only in the IT sector.

Secondly, online lectures are made available by schools on a limited scale due to time, physical, and instructor availability constraints. Moreover, students are shifting to online learning due to its ease and simplicity. During the pandemic, it was challenging for students to execute coding practicals. So, by doing this research, we mainly focus on this problem and the implementation of a must-needed resolution to overcome it. We will achieve it by creating a web portal for students to perform programming practicals. Lastly, our web application will allow instructors to track the progress and do the evaluation.

II. LITERATURE REVIEW

M. Goldman [1] in the paper Role-Based Interfaces for Collaborative Software Development helped instructors to track students' progress without asking them and working with them to improve the area where students are facing issue immediately by checking their code.

L. Murphy et al [2] in the paper Active and Cooperative Learning Techniques for the Computer Science Classroom developed a forum to discuss the challenges and benefits of active and cooperative learning and techniques for incorporating active and cooperative learning into the computer science classroom.

©IJRASET: All Rights are Reserved

S. Ghorashi and C. Jensen [3] in the paper Integrating Collaborative and Live Coding for Distance Education designed and tested a feature-rich collaborative environment for an online class and found that it enhanced learning through methods such as remote pair programming, live coding, and a tight code-to-artifact feedback loop.

L.L. Werner et al [4] in the paper Pair-Programming Helps Female Computer Science Students highlighted how collaborative nature of pair-programming teaches women students that software development is not the competitive, socially isolating activity that they imagined. It encourages women to pursue computer science as a major and as a potential career. Because of this, they strongly advocate the use of pair-programming in all introductory programming courses.

They used pair-programming in all introductory programming courses. Additionally, used optional pair- programming in all upper-division programming courses. The teachers who experimented with pair-programming for the secondary study all strongly believe in it and encourage their students to use it.

J. Lautamäki et al [5] in the paper CoRED: browser-based Collaborative Real-time Editor for Java web applications created CoRED, that is, a complete Java editor with error checking and automatic code generation capabilities, extended with some features commonly associated with social media.

As a proof of the concept, we have extended CoRED to support Java based Vaadin framework for web applications. Moreover, CoRED can be used either as a stand-alone version or as a component of any other software. It is already used as a part of browser based Arvue IDE.

Wu et al [6] in the paper TryLinks: An interactive tutorial system for a cross-tier Web programming language created Links, that is, a web programming language under development in Edinburgh aimed at simplifying web development. This paper reports on a Web-based "TryLinks" system, allowing anyone to experiment with Links without going through the installation process. TryLinks was designed with two major functionalities: an interactive Links shell that teaches the basic syntax of Links and acts as a playground, as well as a short tutorial series on how Links is used in practical web development. Tutorials can also be created or modified by administrators.

We present the design and implementation of TryLinks, and conclude with discussion of lessons learned from this project and remaining challenges for Web-based tutorials for Web programming languages.

B. Nédelec et al [10] in the paper CRATE: Writing Stories Together with our Browsers developed CRATE, that is, a real-time decentralized collaborative editor that runs directly in web browsers thanks to WebRTC. As compared to state-of-the-art, CRATE is the first real-time editor that only requires browsers in order to support collaborative editing and to transparently handle from small to large groups of users. Consequently, CRATE can also be used in massive online lectures, TV shows or large conferences to allow users to share their notes. CRATE's properties rely on two scientific results:

(i) a replicated sequence structure with sub-linear upper bound on space complexity; this prevents the editor from running costly distributed garbage collectors,

(ii) an adaptive peer sampling protocol; this prevent the editor from over-sizing routing tables, hence from letting small networks pay the price of large networks. This paper describes CRATE, its properties and its usage.

T. Jungnickel and T. Herb [13] in the paper Simultaneous editing of JSON objects via operational transformation made a real-time collaboration among geographically distributed users is becoming increasingly important in our private and professional lives. Operational Transformation (OT) is a well established technique for consistency maintenance in the domain of collaborative text editing.

In this paper we expand the scope of OT to support simultaneous editing of shared JSON objects. The presented approach can be used in almost any modern web framework to implement collaboration tools that are, in contrast to popular real-time collaboration tools like Google Docs, no longer restricted to text documents.

III.MOTIVATION

Our application can be used by both teachers as well as students. We created this online coding platform to provide the following advantages to its users:-

1) Firstly, it will make way for practicals along with theoretical learning.
2) Moreover, it will be beneficial for the student community.
3) Also, it will help teachers to conduct practicals extensively.
4) It will keep track of the progress of the student.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue XI Nov 2022- Available at www.ijraset.com

IV. METHODOLOGY

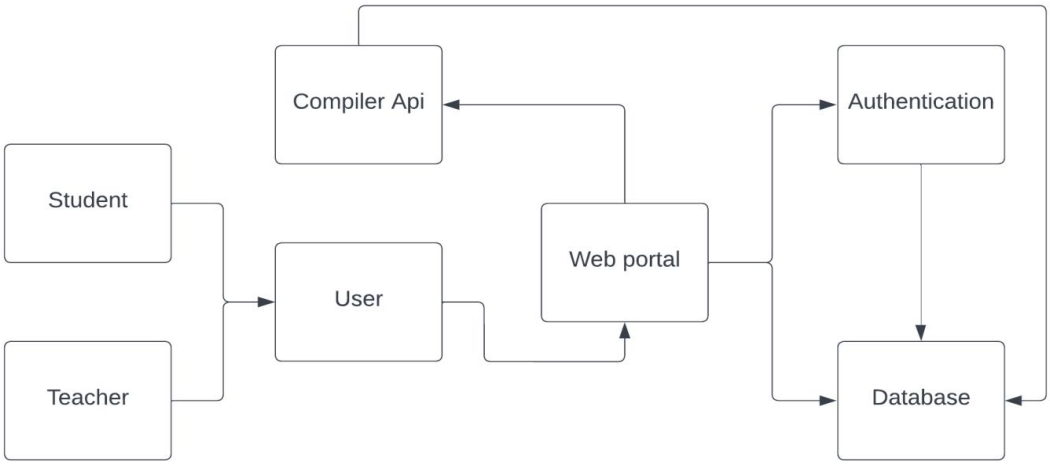


Fig. 4.1. Block diagram

There are mainly two categories of users of our application:- Students and teachers. They are provided a user-friendly GUI in the web portal. Moreover, the web portal uses the compiler API to provide an integrated development environment to its users.
Also, the database is for tracking the progress of students. Lastly, the authentication is maintained to provide a highly secure and error-less platform.

V. PROPOSED WORK

Studying available platforms we came to the conclusion that there was a consistent scarcity of online coding platforms for schools. During the pandemic, it became really hard for students to attend the programming practicals physically. Thus students started performing the practicals on their devices which made it harder for teachers to monitor their practical activities. Thus we are proposing a system where this whole process of programming practicals, their automatic evaluation, and work tracking and management will be placed online. The system involves a web portal where the teachers and students can create their user accounts. Once create both teachers and students will be provided with their specific features. The teachers have features like creating new classes, creating new questions for those classes as well as viewing submission data regarding students' practicals. The students can join particular classes created by the teachers and track their work. When a student selects an uncompleted programming assignment they will have a programming editor with all languages to choose from and on this editor, they can perform their coding then the students can execute their programs then the web portal uses the API to compile the compile and evaluate the program and return its information back to the portal. The portal will then pass this information to be stored in the database so the teacher can monitor it. This way the process will be much easier for teachers and students and much more accessible as well.

1) *Teacher and Student Registration and Login:* Both the teachers and the students can create their accounts using the web portal. The teacher's account will have special privileges compared to the student's. Once the account is created the users can log in back anytime.
2) *Classes Creation and Management:* As stated above the teachers will have special privileges compared to the students. The teacher has the privilege to create a class. The teachers have to assign a password to that class which they can distribute to their class. Then the students can use this password to enroll in this class and perform the assignment. The teachers can also monitor students' work regarding the class.
3) *Code Compilation and API Management:* When the student selects a particular programming assignment from the class he will be provided with an editor to perform his programming operations with multiple language support and a run-time analyzer. Here students can verify the result of their program like the output or how many resources it took. Then the student can submit the assignment and the output information will store in the database which the teachers can monitor.
4) *Database Management:* All the information regarding the users or the classes or the programming submission will be stored in a relational database. The database will interact with the web portal to store information like class information, and student information.

©IJRASET: All Rights are Reserved

5) *Work Tracking:* For students to make sure they are up to date with their submissions we will provide a system that will track their work. So students will be able to keep track of their submissions according to their classes.

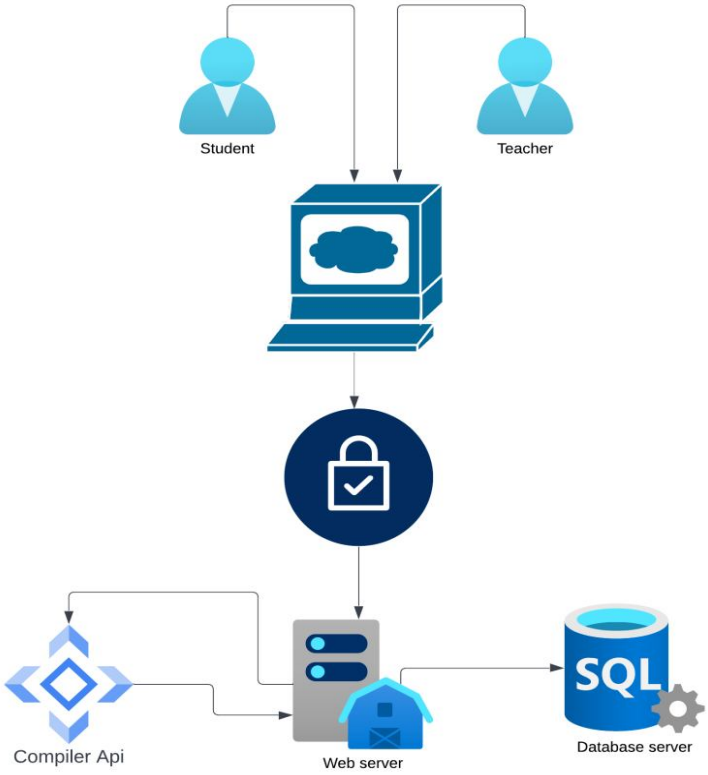


Fig. 5.1. System Architecture

Both students and teachers can interact with the system. It is a highly secure system with proper authentication. The compiler API is used to run the code in the web server. SQL Database server tracking the progress of the student.

VI. RESULTS AND EXPERIMENTATION

The following are the implementation snapshots that demonstrate the step by step working of our application:-

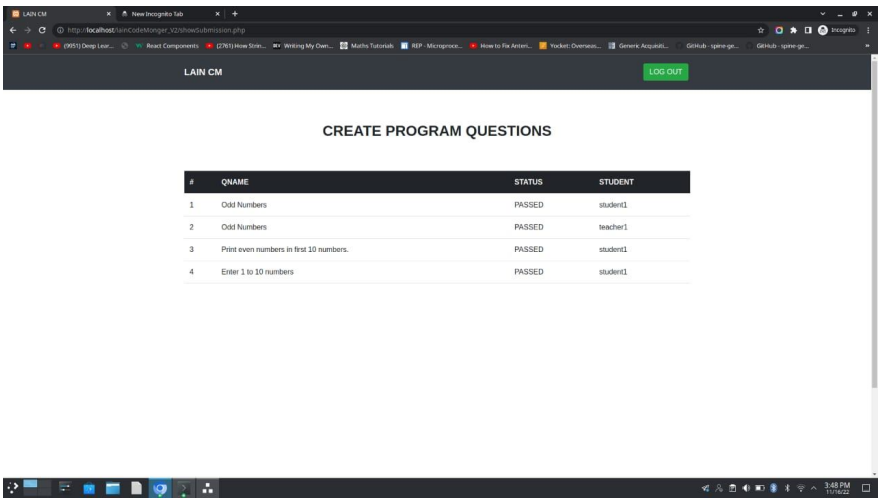


Fig. 6.1. Assign programming questions

Firstly, the faculty can assign programming questions to students and also check the status whether it has been completed or not.

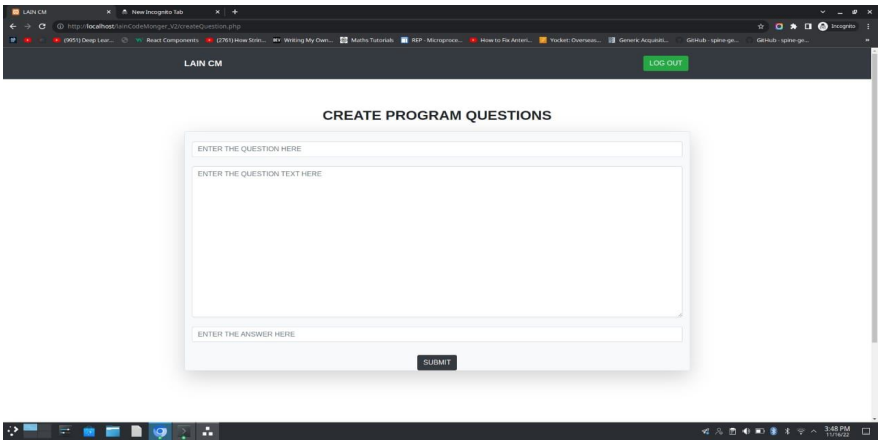


Fig. 6.2. Create programming questions

Also, the faculty can create programming questions by simply entering the question and details with the expected output.

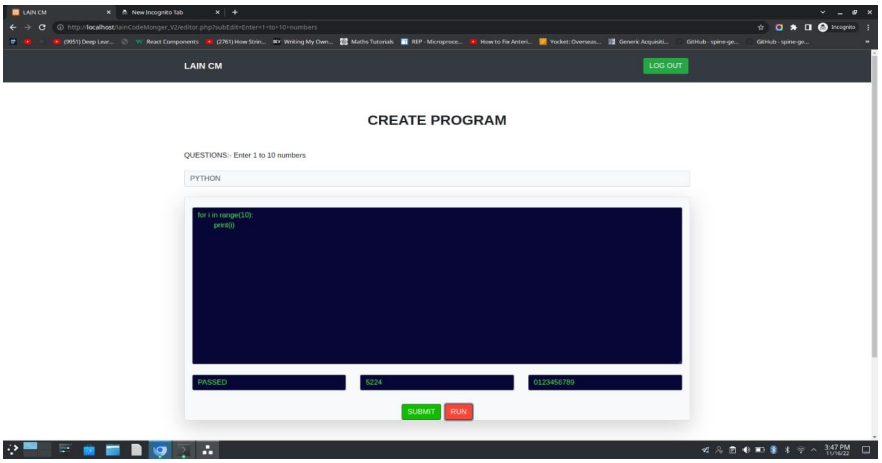


Fig. 6.3. Code editor

The assigned task will appear in the student portal, and students will perform the practical by selecting the language and writing the code in the code editor. it will show if the code was compiled successfully at the bottom of the editor with the output and how much RAM was assigned.

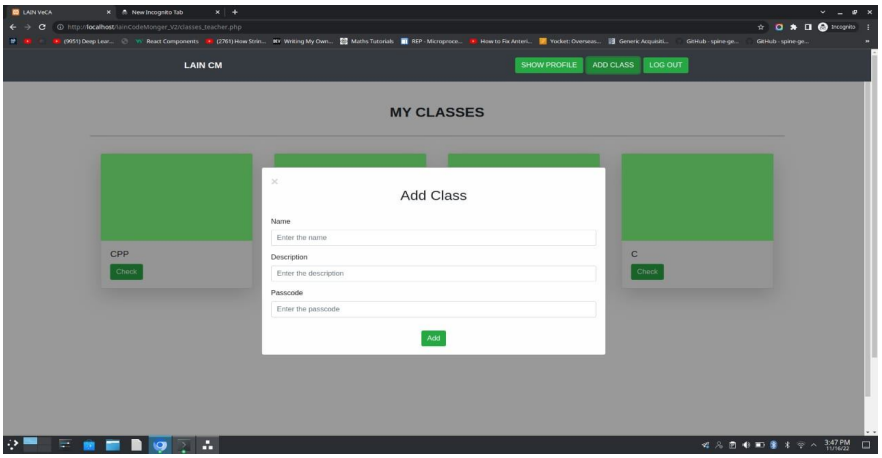


Fig. 6.4. Adding new class

Moreover, in our system, the new class can be added by filling in the necessary details.

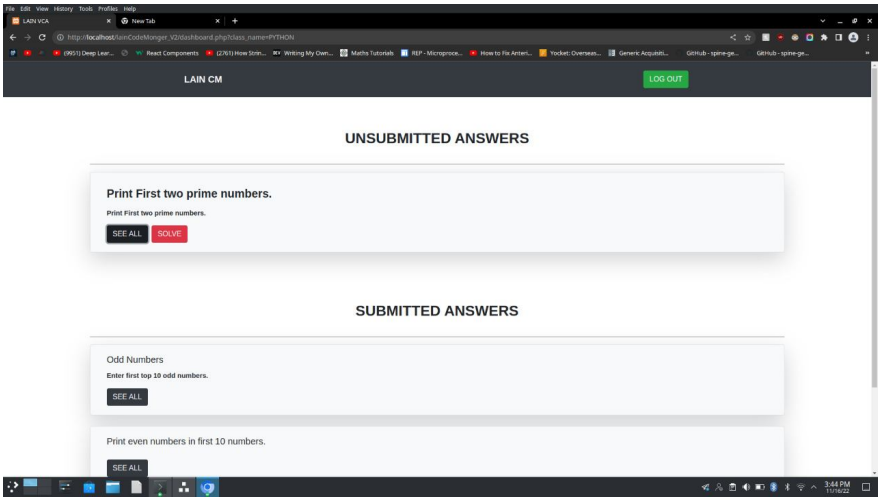


Fig. 6.5. Submitted and unsubmitted answers

Lastly, the faculty and students can check the progress in the portal.

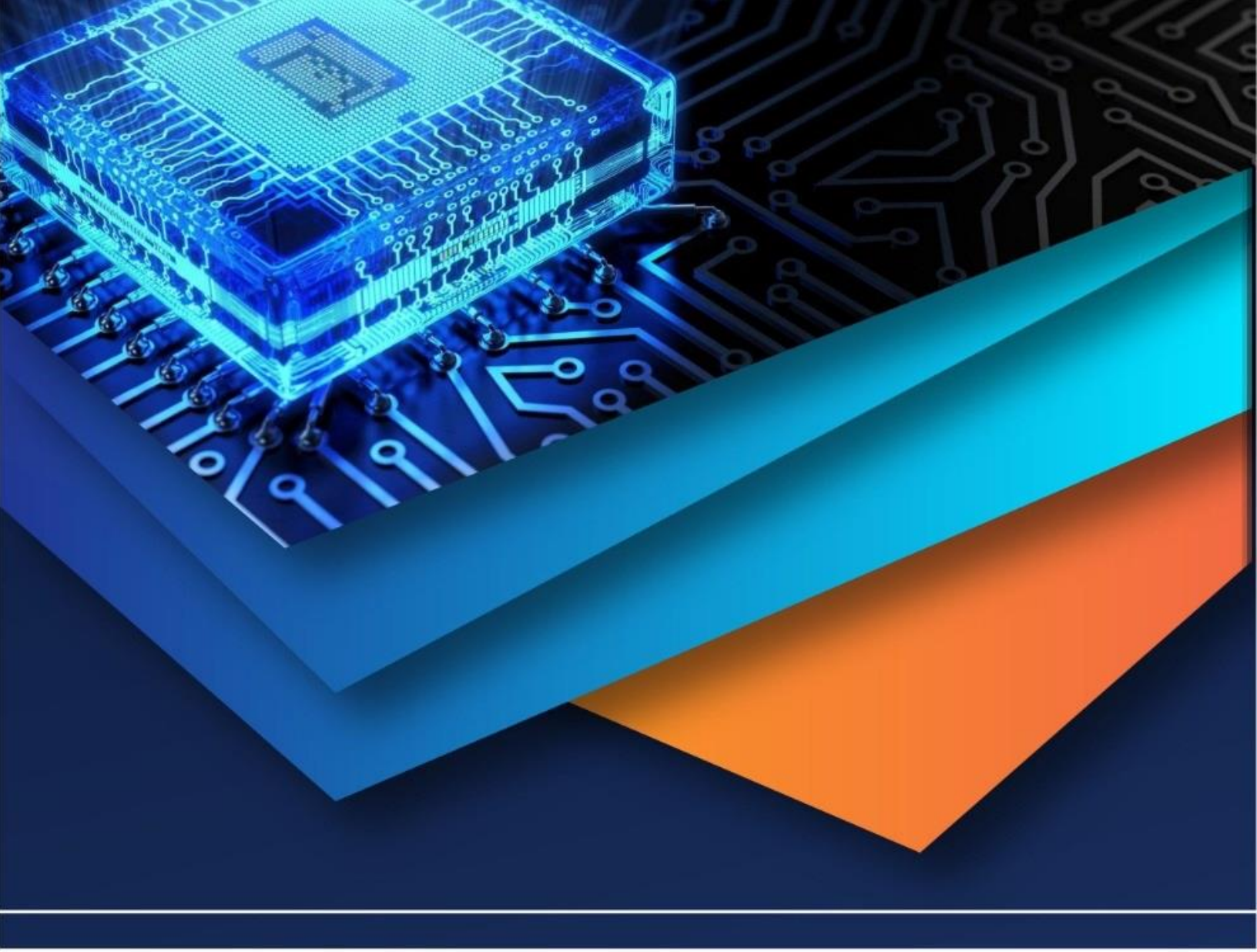
VII. CONCLUSION

In this manner, we created a user-friendly system that is quite beneficial, helping both students and teachers. Here, both of them can perform coding practicals without any hustle. Moreover, it will keep a track of progress of the students. So, it will lead to a digital revolution in the education sector.

Our system takes inspiration from the currently available platforms. It provides an optimal solution to commonly faced issues. By giving a consistent and error-free performance, it has helped many users to start their programming journey.

REFERENCES

[1] M. Goldman, “Role-Based Interfaces for Collaborative Software Development,” Proc.24th ACM Symp. Adjunct User Interface Software and Technology (UIST 11 Adjunct), 2011, pp. 23−26.
[2] L. Murphy et al., “Active and Cooperative Learning Techniques for the Computer Science Classroom,” J. Computing in Small Colleges, vol. 18, no. 2, 2002, pp. 92−94.
[3] S. Ghorashi and C. Jensen, “Integrating Collaborative and Live Coding for Distance Education,” Computer, vol. 50, no. 5, pp. 27–35, May 2017.
[4] L.L. Werner, B. Hanks, C. McDowell, "Pair-Programming Helps Female Computer Science Students", J. Educational Resources, vol. 4, no. 1, 2004.
[5] J. Lautamäki, A. Nieminen, J. Koskinen, T. Aho, T. Mikkonen, and M. Englund, “CoRED: browser-based Collaborative Real-time Editor for Java web applications,” Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work - CSCW 12, Feb. 2012.
[6] Wu, Junao, Arek Mikolajczak, and James Cheney. "TryLinks: An interactive tutorial system for a cross-tier Web programming language." arXiv preprint arXiv:1905.02033 (2019).
[7] “Firepad Integration Documentation,” Firepad. [Online]. Available: https://firepad.io/docs/. [Accessed: 07-Jul-2019].
[8] Yudai, “yudai/gotty,” GitHub, 13-Dec-2017. [Online]. Available: https://github.com/yudai/gotty. [Accessed: 28-Jul-2019].
[9] Ajaxorg, “ajaxorg/ace,” GitHub, 03-Dec-2016. [Online]. Available: https://github.com/ajaxorg/ace. [Accessed: 13-Dec-2019].
[10] B. Nédelec, P. Molli, and A. Mostefaoui, “CRATE: Writing Stories Together with our Browsers,” Proceedings of the 25th International Conference Companion on World Wide Web - WWW 16 Companion, pp. 231–234, Apr. 2016.
[11] M. Lehenbauer, “Announcing Firepad - Our Open Source Colla borative Text Editor,” The Firebase Blog, 09-Apr-2013. [Online]. Available: https://firebase.googleblog.com/2013/04/announcing-firepad-our-open- source.html. [Accessed: 21-Aug-2019].
[12] “Operational transformation,” Wikipedia, 06-Oct-2019. [Online]. Available: https://en.wikipedia.org/wiki/Operational_transformation. [Accessed: 15-Oct-2019].
[13] T. Jungnickel and T. Herb, “Simultaneous editing of JSON objects via operational transformation,” Proceedings of the 31st Annual ACM Symposium on Applied Computing - SAC 16, pp. 812–815, Apr. 2016.
[14] “Read–eval–print loop,” Wikipedia, 30-May-2019. [Online]. Available: https://en.wikipedia.org/wiki/Read–eval–print_loop. [Accessed: 13-Sep-2019].
[15] A. Masad, “Making Repl.it Collaborative at Heart,” repl.it, 18-Sep-2019. [Online]. Available: https://repl.it/site/blog/collab. [Accessed: 20-Sep-2019].
[16] R. Sharrock, L. Angrave, and E. Hamonic, “WebLinux,” Proceedings of the Fifth Annual ACM Conference on Learning at Scale - L@S 18, Jun. 2018.
[17] L. Eggleston, “2019 Coding Bootcamp Market Size Study,” Course Report, 01- Aug-2019. [Online]. Available: https://www.coursereport.com/reports/coding-bootcamp-market-size-research-2019. [Accessed: 10-Aug-2019].



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)