

# V-Code: Online Code Editor

<sup>1</sup>Prof. Poonam R. Pathak, <sup>2</sup>Tejas V. Magade, <sup>3</sup>Avishkar A. Vichare, <sup>4</sup>Shreyas I. Repale

<sup>1</sup>Professor, Department of Information  
Technology, Pillai HOC College of  
Engineering and Technology,

<sup>2,3,4</sup>Student,

<sup>2,3,4</sup> Department of Information Technology,  
Pillai HOC College of Engineering and  
Technology, Maharashtra, India,

**Abstract**— V-Code is an online collaborative code editor designed for remote software development teams. It features real-time collaboration, Compilation & Execution of code and gives suggestions regarding errors, collaboration using SOA between client and server through collaborator tools, error suggestion through Open API's machine learning techniques. The system is built using web technologies such as Nuxt.js on the client side, and Node.js, Python Flask and Express on the server side. V-Code has the potential to improve the productivity of code editor teams and it also represents a significant contribution of collaborative effort towards rectifying code errors.

**Keywords:** Collaborative Software Development Team, Online Code Editor, Open API, Error Suggestion

## I. INTRODUCTION

Our system aims to cater to the needs of coding enthusiasts of all levels, from beginners to professionals, and facilitate the learning process by providing a range of features. These features include live code sharing, a playground for multiple programming languages, a web development playground, and an online community for coders. The live code sharing feature allows users to collaborate on code in real-time, which is especially useful for remote team collaboration. The playground offers an environment for users to practice programming languages like Java, Python, Ruby, C, C++, and more. Finally, the online community provides a platform for users to connect with other coders and solve errors in their coding systems. Overall, our system aims to make coding more accessible, collaborative, and enjoyable for coders of all levels.

## II. SCOPE

The system aims to develop a web-based application that offers a collaborative editor, compiler, and execution environment for programming languages. The system will be accessible to users without the need to download or install any software on their local machines. The collaborative editor will support multiple programming languages and provide various features. The compiler will be integrated with the editor and provide real-time feedback to users on any syntax errors or other issues. Additionally, the system will include an execution environment for users to run and test their code. Overall, the system seeks to provide a seamless and accessible programming environment for users of all levels, making it easier for them to collaborate and work on code together.

## III. PROBLEM STATEMENT

As coders, we often encounter errors or unexpected outputs in our code, and we typically resort to sharing our screen on a video call or sending screenshots of our code, or even browsing to find solutions to the bugs in our code. The system we are building will not only allow us to share our code, but also invite collaborators to edit it in real-time. Any changes made on the expert's end will automatically reflect on the client's side as well. This live code sharing and editing feature is sure to assist aspiring coders in resolving issues in their code.

## IV. RELATED WORK

Online collaborative code editors have become increasingly popular in recent years, as they allow remote teams to work together on coding systems in real-time. Some popular examples include Codepen, JSFiddle, Repl.it, and Programiz. Codepen is an online code editor that allows users to write, test, and share HTML, CSS, and JavaScript code snippets. Repl.it is also a popular online code editor that provides users with a wide range of programming languages to compile and execute.

### 4.1 Codepen

Codepen is an online front-end development tool that allows users to create, test, and share HTML, CSS, and JavaScript code snippets. It was launched in 2012 and has since become a go-to tool for web developers and designers. Codepen offers a range of features, including a live preview of code changes, and the ability to browse and fork other users' code snippets.

## 4.2 Repl.it

Repl.it is an online development environment that allows coders to write, compile, and execute code in various programming languages. The platform provides a range of features, including a code editor, a debugger, and a community for coders.

## 4.3 JSFiddle

JSFiddle is an online code editor and development tool that allows users to write, edit, and test their JavaScript, HTML, and CSS code in real-time. It features a user-friendly interface that makes it easy to create and share code snippets and web pages with others.

## V. IMPLEMENTATION

### A. methodology

The V-Code system has three main modules. Module 1 is the Live Code Share feature, which provides real-time code sharing with other parties. Module 2 is Code Execution and Compilation, which executes code and shows the desired output. Module 3 is Error Suggestion, where collaborators (experts) not only offer solutions but also provide helpful suggestions to coders.

### 5.1 Module 1: Collaborative Feature

The Live Code Share feature [1] is a collaborative tool designed to assist programmers. With this feature, users can create a room and invite others to join by sharing code. When a user joins a room, they can see and edit all the code. Any changes made by a user will be reflected in real-time on the other side thanks to the [Socket.io](#) [7] real-time connection. Multiple users can join a single room. We used Socket.io to implement this feature. Socket.io is a JavaScript library that enables real-time bidirectional connections between web clients and servers. It allows the server to push data to clients as soon as it becomes available, rather than waiting for requests from the clients. We have hosted it on [glitch.io](#).

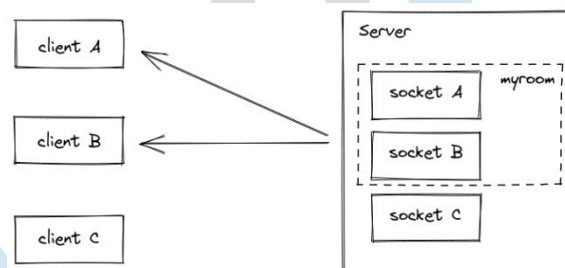


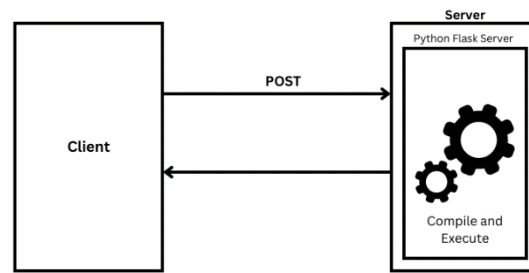
Fig.1 Socket.io Rooms Workflow

Socket.io provides the functionality to create virtual channels, called "rooms," which allow clients to join and leave as needed. When a client joins a room, they receive all events that are emitted to that particular room. If a client sends an event, they can direct it to a specific room by specifying the room name, and only clients who have joined that particular room will receive the event.

Socket.io provides the functionality to create virtual channels, called "rooms," which allow clients to join and leave as needed. When a client joins a room, they receive all events that are emitted to that particular room. If a client sends an event, they can direct it to a specific room by specifying the room name, and only clients who have joined that particular room will receive the event. Rooms can be created either by the server or by the client. If a client creates a room, they will automatically join that room. The joining process can be confirmed with a callback function. Similarly, when a client leaves a room, they can specify a callback function to confirm that they have successfully left the room. [1][7].

### 5.2 Module 2: Code Compilation and Execution

This feature [2] allows users to execute almost all programming languages without having to download or install them on their own systems. We built this feature using Python Flask [8] and hosted it on the [Oracle Cloud](#). When a user clicks on the "Run" button, a request is sent from the frontend to our Oracle Cloud. The code file is then created on the cloud, and the code runs there. The output generated is captured and sent back to the frontend, where it is displayed to the user [3][4].



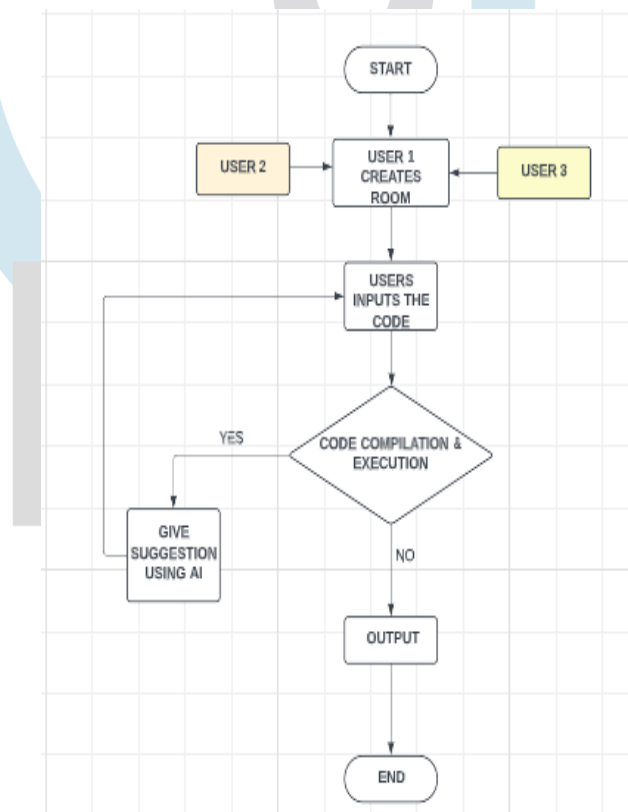
**Fig2.** SOA (Service-Oriented Architecture)

### 5.3 Module 3: Error Suggestion

To improve the user experience of our online code compiler, we integrated an [OpenAI API](#) [6] that offers error suggestions for code. This API analyzes the syntax of the code and recognizes any potential errors or issues that may arise. By utilizing this API, we can assist users in quickly detecting and correcting any problems with their code, ensuring that their programs operate smoothly and efficiently. The OpenAI API uses machine learning algorithms to evaluate and interpret code, providing highly accurate error suggestions and identifying potential issues that the user may have overlooked. The API is also user-friendly and intuitive, making it simple for users of all skill levels to understand the suggestions and implement the necessary corrections to their code.

By using the OpenAI API [6], we guarantee that our online code compiler is a potent and efficient tool for programmers of all skill levels. Our compiler offers a streamlined and efficient means of writing, editing, and testing code, allowing users to accomplish their programming objectives with ease and confidence, whether they are seasoned developers or just starting out.

## VI. SYSTEM DESIGN



**Fig.3** System Flow Diagram

## VII. RESULT DISCUSSION

To enhance the coding experience for users, we developed V-Code, an online code editor with four key features: a live code share function, an online compiler supporting almost all programming languages, a web dev playground, and a community for developers. Our live code share function enables real-time code sharing and editing through Socket.io. To simplify the coding process, our system provides an online compiler, eliminating the need to download or install programming languages. We also incorporated OpenAI's error suggestion API to promptly identify and correct any coding issues. Finally, we hosted V-Code on Glitch.io, Oracle Cloud and [Netlify](#) for reliability and scalability.

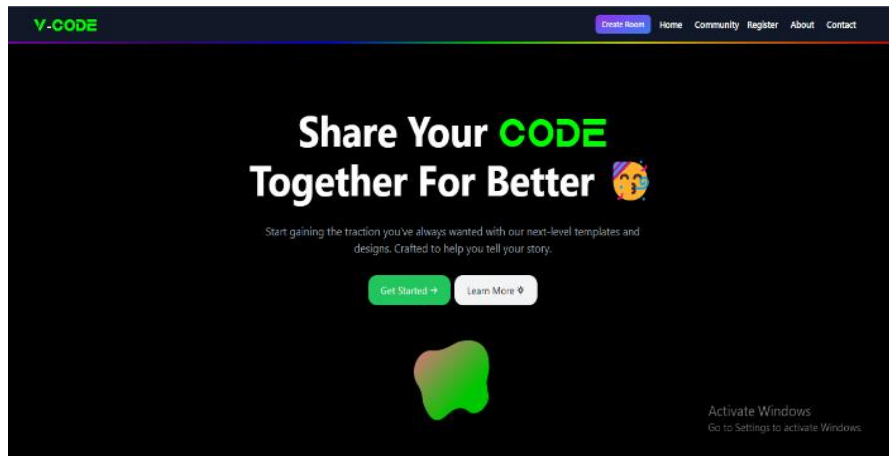


Fig. 4 Home Page

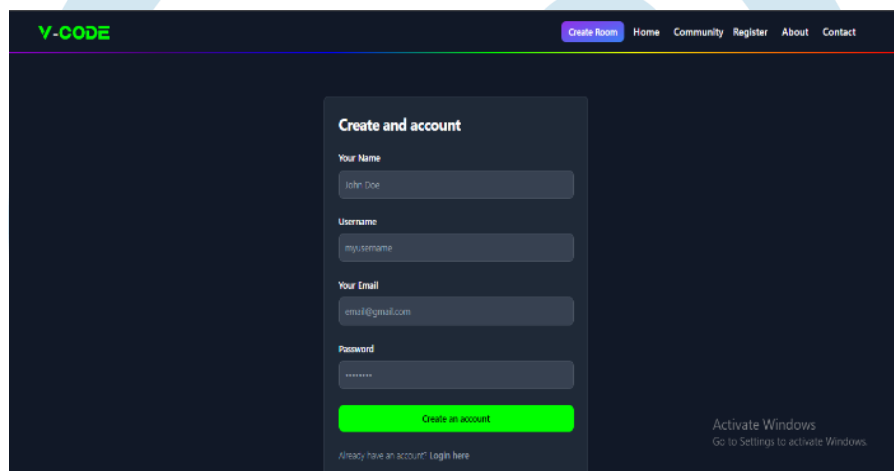


Fig. 5 Authentication Page

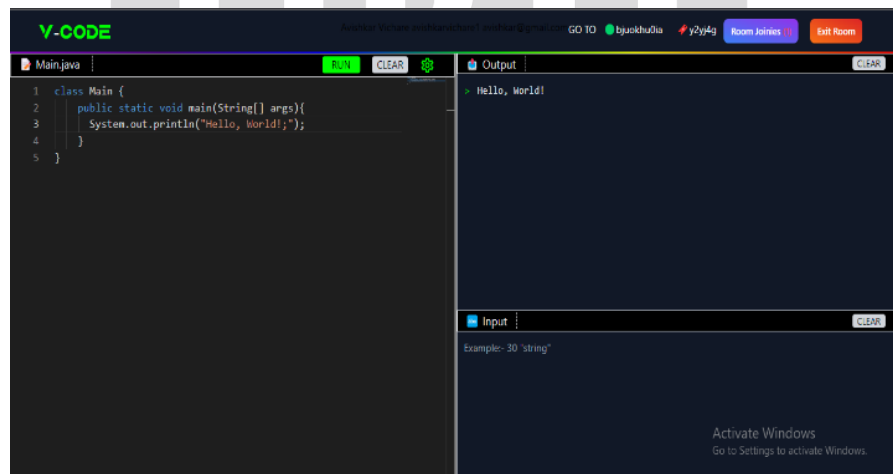


Fig. 6 Code Editor Page

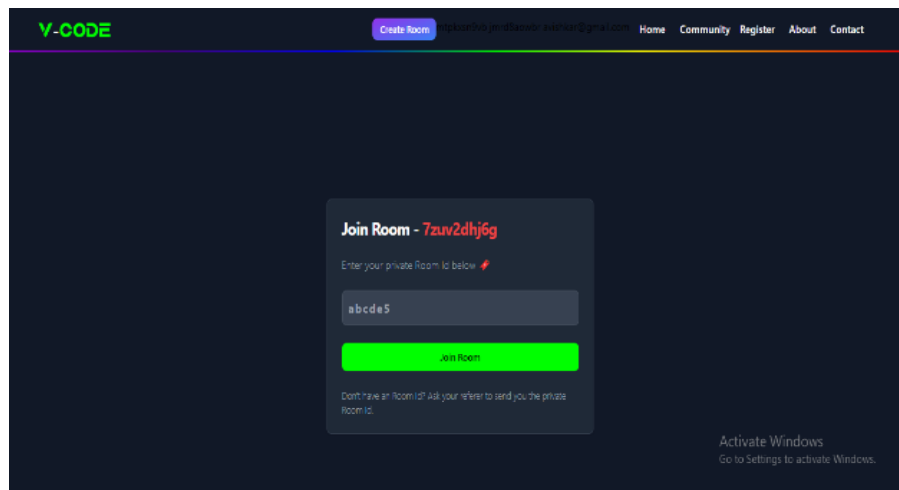


Fig. 7 Room Joining Form Page\

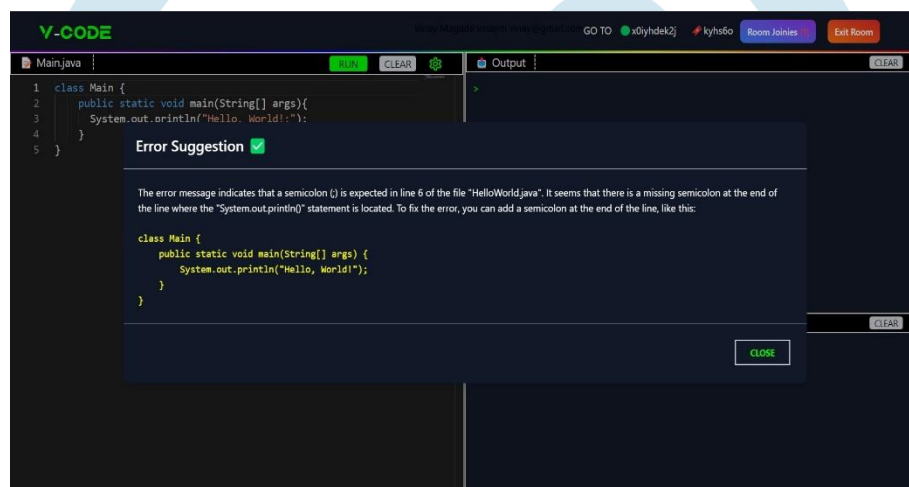


Fig.8 Error Suggestion Page

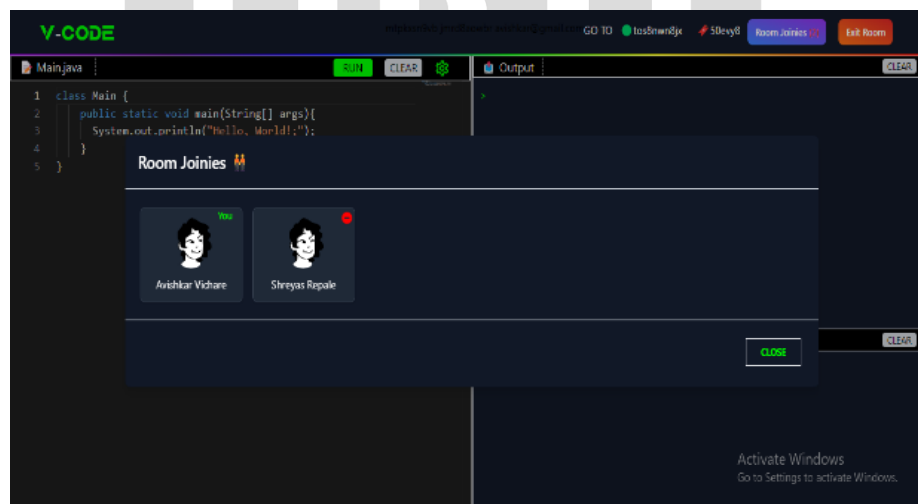


Fig.9 Room Joiners Page

## VIII. LIMITATIONS OF THE WORK

Although our system has various strengths and advantages, it is important to acknowledge certain limitations and areas that require improvement. One of the limitations of our live code share feature in its current version is that it may have scalability and performance issues when too many users are connected simultaneously, affecting its reliability. While the OpenAPI error suggestion API is a helpful tool, it is not infallible and may occasionally give inaccurate or irrelevant suggestions, which users should verify and correct themselves. At last, the system does not support some specialized tools, frameworks and languages for error removal. Future work can be focused on addressing these limitations and expanding our system capabilities to cater the

needs and scope of developers.

## IX. FUTURE WORK

To continue enhancing our system, future work includes improving the scalability and performance of our live code share feature, enhancing the accuracy and relevance of the error suggestion API, and expanding support for specialized programming languages and frameworks. We also plan to gather user feedback through testing and continually improve the user experience while addressing any issues or limitations.

## X. CONCLUSION

V-Code is an innovative tool designed to support remote software development teams. With its real-time collaboration features, code compilation and execution, and error suggestion using OpenAI's machine learning algorithms and collaboration tools, it has the potential to significantly improve the productivity of code editor teams. It uses technologies such as Nuxt.js, Node.js, Python Flask, and Express.

Its features include real-time collaboration and error correction suggestions, making it an inclusive space where individuals from different backgrounds and skill levels can come together to learn, share knowledge, and improve their coding abilities. This contribution to the field of online code editing is significant as it offers a convenient and efficient platform for remote teams to work together and streamline their coding processes.

## REFERENCES

- [1] Y. Singh, H. Singh, J. Chauhan, "Online Collaborative Text Editor Using Socket.IO", 21666312  
<https://ieeexplore.ieee.org/document/9725782>
- [2] W. Kimpan, T. Meebunrot, B. Sricharoen, "Online code editor on Private cloud computing", 6694748  
<https://ieeexplore.ieee.org/document/6694748>
- [3] A. Ansari, S. Patil, A. Navada, A. Peshave, V. Borole, "Online C/C++ compiler using cloud computing", 6002124  
<https://ieeexplore.ieee.org/document/6002124>
- [4] H. Došilović, I. Mekterović, "Robust and Scalable Online Code Execution System", 9245310  
<https://ieeexplore.ieee.org/document/9245310>
- [5] F. Mulla, S. Nair, A. Chhabria, "Cross system C Compiler", 7859982  
<https://ieeexplore.ieee.org/document/7859982>
- [6] OpenAI API documentation. (n.d.). Retrieved from <https://beta.openai.com/docs/api-reference>
- [7] Nuxt.js Documentation. (n. d). Retrieved from <https://nuxt.com/docs>
- [8] Socket.IO documentation. (n.d.). Retrieved from <https://socket.io/docs/>
- [9] Flask documentation. (n.d.). Retrieved from <https://flask.palletsystem s.com/en/2.1.x/>
- [10] Oracle Cloud documentation. (n.d.). Retrieved from <https://docs.oracle.com/en/cloud/get-starte/index.html>