

## **CODESPHERE: Code Editor**

J. Naveen<sup>1</sup>, Sadiya Mazarath<sup>2</sup>, V.Rajeshwari<sup>3</sup>, T.Bhargavi<sup>4</sup>, Syed Raiyan<sup>5</sup>, J.Chaitanya<sup>6</sup>

<sup>2,3,4,5</sup>BTech Students, Department of CSE, Balaji Institute of Technology and Science, Laknepally, Warangal, India

<sup>1</sup>Project Guide, Department of CSE, Balaji Institute of Technology & Science, Laknepally, Warangal, India

<sup>6</sup>Assistant Professor, Department of CSE, Balaji Institute of Technology & Science, Laknepally, Warangal, India

### **ABSTRACT**

CodeSphere is an online code editor that works with HTML, Python, and JavaScript. It has a special Voice-To-Code function that lets users write code with voice commands for a convenient and hands-free experience. Additionally, CodeSphere offers a Save Locally feature that enables users to store their projects locally on their devices. You can use the editor without a login or with one to safely store your work. CodeSphere is ideal for programmers of all skill levels due to its user-friendly interface, which provides a platform for learning, practicing, and enhancing coding abilities. By taking away a crucial tool for novice developers.

An online code editor is a web-based tool that enables users to write, edit, compile, and run codes right in the browser, thus eliminating the need for local installations of Integrated Development Environments (IDEs). The tools support a variety of programming languages and often come with options such as syntax highlighting, error checking, version control integration, real-time collaboration, and cloud storage for code. Online code editors are built on cloud infrastructure that provides accessibility for developers to work from any device with internet access.

They greatly enhance collaboration among teams and offer a handy learning and experimentation environment for beginners. In tandem with the cloud computing boom, editors become highly scalable, cost-efficient, and easily deployed options for independent programmers and corporations alike. As web technology continues to evolve rapidly, online code editors are becoming more and more powerful and versatile, thus effectively bridging the gap with traditional desktop-based systems.

**Keywords:** IDE, developers, code, and code editor.

### **1.INTRODUCTION**

Recently, there has been a significant shift in the programming industry. As the demand for software development has increased, so too has the need for efficient, affordable, and cooperative programming tools. One such tool that has become increasingly popular is the online code editor.

With an online code editor, which is a web-based platform that does not require the installation of any software on the user's computer, programmers can write, edit, and test their code in a browser. Developers can now coordinate efforts, share code, and work together remotely more quickly.

Online code editors' features, like syntax highlighting, autocomplete, and error-handling tools, may appeal to developers. Code editors are software tools designed to make it easier for developers to write, edit, and save their code.

There are numerous options for code editors, ranging from basic editors to fully functional Integrated Development Environments. Among the most popular code editors are Notepad++, Atom, Visual Studio Code, and Sublime Text.

Code editors make the task of building better programs very dynamic for the developer. It increases the speed of developing and debugging the code while providing tools for easy handling of a big code base. These editors are custom-made through plugins and extensions to suit the specific needs of a given developer or project[1-28].

## **2. OBJECTIVE**

The online code editing feature is meant to facilitate a cloud-based, available and synchronized platform for the functioning of a developer who can write, edit, debug, and even test code in different languages at any time on any device. Some of the major objectives include:

- Multi-language Support of several programming languages (for example, JavaScript, Python, HTML).
- Real-Time Collaboration by which team members can edit codes at the same moment.
- Cloud-based Storage, thus easy accessibility, version control, and seamless file management.
- Perform implementation and test directly inside the editor for immediate feedback.
- Error Detection and Debugging tools to facilitate faster identification and elimination of issues.
- User-friendly Interface which boosts productivity and minimizes setting up.
- Security and Privacy of code through the use of encrypted connections and access controls to it.
- Integration with External Tools like git and CI/CD pipelines to streamline workflow.

The purpose of creating such objectives is to ensure a productive, accessible, and collaborative development atmosphere for coding practices as an individual or as a team.

### **3. LITERATURE SURVEY**

#### **A. Online code editor on Private cloud computing**

As clarified in this paper, the Online Code Editor will cater for programmers or developers who are willing to create applications without platform constraints or certain specific physical requirements. It runs entirely as a web application off a private cloud. The editor is powered by HTML, PHP, CSS, and JavaScript among many other web programming languages.

The editor identifies different languages of programming by the highlight of syntax of the programs. Users can even create new server projects and files, import files, and export files as well. More options include Save, Auto Save, Delete and more. "Ace", the open source technology, was used in this study in the development of the text editor for such features like Undo, Redo and Syntax highlight.

Experimental results indicate that the proposed editor is appropriate for use with private cloud computing. A comparison of the capability of desktop PC-based editors Notepad++ and EditPlus was also given.

#### **B. Review on Server Based Code Editor**

This paper demonstrates a server-based code editor for Java code. Some code editors have the ability to run JavaScript, CSS, and HTML.

Programmers are hampered with difficulties when working on programming languages such as Java, PHP, etc. Whereas Java is a new addition to this project, it would be the one used by programmers to run in a server-based code editor.

It is a proper use of the ability to run multiple programming languages in one platform.

#### **C. CodeR: Real-time Code Editor Application for Collaborative Programming**

The goal of this research is to develop a real-time code editor application over web socket technology so that collaboration can take place among users during the working project. This application has a feature that forms a platform for users to collaborate on a project in real time. The researchers are using analytical methods in studying some of the current code editor programs by conducting surveys and reviewing the already available literature. Users are given access to a workspace where they can write code, run it, observe the terminal results and interact in real-time with other users. This is enabled via a web service called CodeR.

## D. BROWSER BASED CODE EDITOR

The authors are proposing to create a website where users could write C codes, copy them, paste them to the C-code editor, and press the run button. The system will then send the information to the server. So, instead of going through the tedious installation of the entire C compiler along with DOS, which turns out to be a pain and an entirely waste of time, this application will help to save time. Therefore, the online compiler finds direct applications to the end-user, which is easy and fast

### 4. EXISTING SYSTEM

These platforms and many other emerging platforms and technologies that support multiple languages and voice-based coding are not yet fully developed mainstream online code editors offering JavaScript, Python, and HTML editing-all-in-one code editing with voice-to-code functionality. Below is a rundown of an existing system offering those capabilities:

#### 1. Replit

- **Features:** Replit offers a lot of programming languages, including JavaScript, Python, and many more, in addition to HTML.
- **Limitations:** It lacks a true, powerful, built-in voice-to-code capability; integration via third-party tools becomes quite tedious.

#### 2. VoiceCode

- **Features:** It has the capability to transcribe voice commands into code syntax so that coding might be performed hands-free.
- **Limitations:** It won't give an accurate speech-to-text transcription report, especially during complex coding. Misrealizations of syntax could happen that might lead to errors.

#### 3. CoderPad

- **Features:** Collaboration allows multiple users to see the changes to the actual code as they code in the tools simultaneously; this works wonderfully with pair programming or group projects.
- **Limitations:** Project management features in CoderPad are weak compared to its competitors in cloud IDEs, for example, Replit.

#### 4. Google Colab

- **Features:** All Python functionality enhanced by a rich library ranging from TensorFlow, PyTorch, etc.

- **Limitations:** Being a cloud application, quality and accessibility rely upon internet connection speed.

## 5. GitHub Codespaces

- **Features:** Enables several users to work on a project collaboratively while in real time coding.
- **Limitations:** Complex Setup for Voice: Technical know-how is necessary to set up external APIs or tools for voice coding.

## 5. PROPOSED SYSTEM

The system we propose is an online code editor acting as an integrated development environment (IDE) that combines full support for JavaScript, Python, and HTML with its innovative voice-to-code capabilities. The proposed system allows developers to code using traditional keyboard inputs as well as voice commands. It is an empowering tool for the disabled, an efficient tool of choice for many, or even a collaborative environment for a team to engage in development.

•**Multilanguage Support:** Supports JavaScript, Python, and HTML for web and backend development.

•**Voice-to-Code Functionality:** This recognizes voice commands for coding actions, such as writing functions, making variables, or moving across code. Engine for speech to text for transcribing code-is spoken directly into the editor.

•**Real-Time Collaboration:** Different users can be editing the same code base at the same time with the changes visible to them instantly.

•**Cloud-based File Management:** Code saved automatically in the cloud shall ensure access via any device.

•**Code Execution:** Live execution of JavaScript, Python, and HTML code on platform.

•**Customizable Voice Commands:** Users can create custom voice commands for repetitive coding tasks (e.g., "Insert Loop", "Define Variable").

•**Real-time Syntax Highlighting and Error Detection:**Helps users by syntax highlighting and instant feedback on error detection while users dictate the code.

### ADVANTAGES OF THE PROPOSED SYSTEM:

**Accessibility Across Devices:**Web-based editors are free from the hassles of local installation and can be accessed from any device with an internet connection.

**Instant Feedback:** Since code is executed in real time, results emerge faster, enabling developers to debug more effectively.

**Cross-Platform Compatibility:** The editor runs on various operating systems and devices to allow for seamless transitions between them.

**Voice-to-Code Conversion:** Developers can code hands-free by converting their voice commands into code, ensuring accessibility and increasing speed as a result.

**Cloud Storage:** Codes are stored in the cloud for ease of recovery and access on multiple devices without fear of data losses.

## **6.SYSTEM REQUIREMENTS**

### **Software Needs:**

Front-End Web Development Framework: Using React.js or Vue.js for the found interface.

Code Execution Engines: JavaScript is executed by Node.js; the Python runtime serves Python code; embedded browser renders HTML/CSS.

Cloud Storage: AWS S3, Firebase, or Google Cloud Storage will retain and retrieve users' code.

Code Editor Library: CodeMirror or Monaco Editor for syntax highlighting, code folding, and auto-completion.

Backend Server: Node.js supports the APIs, user authentication, and real-time data handling.

Voice-to-code: Includes Google Cloud Speech-to-Text and Web Speech API for voice-to-code.

### **Hardware requirements:**

Server Infrastructure: Scalable Cloud Servers (for instance, AWS EC2, Google Cloud Compute) for code execution and storage.

Database Server: MongoDB or MySQL for recording user accounts, projects, and code files.

Execution Environment: Docker containers for secure executions in isolated environments.

Web Hosting: Reliable web hosting with minimum downtime (for instance, AWS, Netlify, Heroku).

CDN (Content Delivery Network): Use a CDN such as Cloudflare for fast content delivery with less latency.

Back-Up Storage: Cloud-based backup and versioning for user projects and code files.

## **7. PROBLEM STATEMENT**

In the fast-paced environment of software development today, developers are mostly challenged by the management of code and its collaborative nature in real-time. Conventional code editors call for local setups, multiple tools for version control, and at times, provide limited accessibility options. Thus, collaboration has become even harder without an efficient platform across varied teams, more so for those working from remote locations.

Therefore, there exists a unified cloud-based online code editor that supports various programming languages (like JavaScript, Python, HTML, etc.) and gives real-time collaboration, great version control, and live code execution. The platform should also be highly accessible from any device, supporting both individual and team-based development workflows and would then be enhanced with features such as error detection, debugging, and user-oriented interface customization.

The absence of an integrated tool for real-time collaboration, for instant code execution, and for effective project management with the contemporary online code editors does not allow developers to work fast and efficiently, especially in the case of distributed teams. Besides, some online editors don't have advanced features of traditional IDEs, limiting their functionality for professional developers.

Therefore, the problem is to build an online code editor that, apart from coding, allows collaboration in real-time, project management in the cloud, and error-free coding on the same platform, ensuring easy and efficient development by individual programmers and teams.

## **8. METHODOLOGY**

Now let us detail the major steps used in online code editor development methodology:

### **1. Requirements Gathering**

- Core features like syntax highlighting, code execution, version control, user authentication, etc.
- Decide on languages to be supported e.g. JavaScript, Python, HTML, etc.
- Understand user requirements: education, collaboration, or personal projects.

### **2. Architecture Design**

- Frontend Model: User interface and interaction model (HTML, CSS, JavaScript, React, Vue.js, etc).

Editor (with libraries, for example, Monaco, CodeMirror, or Ace).

User interface components include buttons, file management, settings.

- Back End: Server-side logic for handling code execution and user management.

Code execution engines (e.g. with Docker containers or serverless functions).

Authentication (e.g. OAuth, JWT, etc.), project management, storage, etc.

- Database: Store user data, projects, and preferences (SQL or NoSQL).

### **3. Code Execution Environment**

- Create a secure sandbox environment through which code can execute (either using Docker or VMs).

- Isolation of processes preventing vectors for security vulnerability.

- Handle language-specific compilers/interpreters on the backend (e.g., Python, Node.js, etc.).

### **4. Editor Features**

-Syntax Highlighting: Syntax highlighting will use documents like Monaco, CodeMirror, or Ace.

-Autocomplete/IntelliSense:Basic autocompletion of terms will be made according to the language parsing.

-Error Reporting:Real-time checking for syntax errors when the user types.

-Version Control: Implements Git or a simple version-control system for saving and tracking a revision of code.

### **5. Real-Time Collaboration (Optional)**

- Sync multiple users so that they can edit as well as view the code online at the same time (e.g. using WebSockets, Firebase).

- Reach a concurrency control with methods like Operational Transformation (OT) or Conflict-Free Replicated Data Types (CRDTs).

### **6. Security Measures**

- Enact access control to bar any unauthorized execution of a code or leakage of data.
- Provide complete sandboxing for code environment to prevent malicious activity.
- Use encryption both for the data transmission and for user authentication.

## **7. Testing and Debugging**

- The functionality of the editor is going to be continuously tested (unit testing, integration testing).
- Test the different edge cases with various types of code for code execution environment testing.
- The entire system including backend and user data management undergoes security audit.

## **8. Deployment**

- Host the app on a scalable cloud platform, e.g., AWS, Google Cloud, Heroku.
- Use containerization through Docker to make the deployments consistent and portable.
- CI/CD implementation for automated testing and deployment.

## **9. Maintenance and Updates**

- The performance of the editor will be continuously monitored along with error reports.
- Periodical updates with the latest features, bug fixes, and security patches will be rolled out.

It's a high-level model of methodologies for creating an online code editor: security with a high level of performance and smoothness in terms of user experience.

## **9. CONCLUSION**

An online compiler is a powerful tool that allows you to write, compile, and execute codes of various programming languages directly in a web browser without installing any local software. Accessibility, convenience, and flexibility offered by it are ideally suited for beginners, students, or even professionals who just require a quick environment for testing and running codes. However, this might come up with limitations, such as resource constraints, performance constraints, and dependency on internet connectivity. Still, such online compilers remain at the center of real learning, prototyping, and collaborative coding for easy access and development of a community of programmers.

An online compiler is a powerful tool that allows users to write, compile, and execute code in various programming languages directly from a web browser and has nothing to do with

installing any software locally. Accessibility, convenience, and flexibility are the terms that best describe it for a beginner, student, or even a professional requiring a testing and running code environment immediately. Limitations imposed include possibly lower performance, resource constraints, and dependency on the internet. Nevertheless, online compilers still stay on the cutting edge of learning, prototyping, and collaborated coding for easy access and generating a community of programmers.

## **10.FUTURE SCOPE**

In order to further develop our system, upcoming works will include increasing scalability and performance of our live code share feature, increasing the accuracy and relevance of the error suggestion API, and providing additional support for specialized programming languages and frameworks. User feedback will be solicited through testing, which in turn will inform continuous improvements on the user experience, as well as issues or limitations.

## **11. REFERENCES**

1. W. Kimpan, T. Meebunrot and B. Sricharoen, "Online code editor on Private cloud computing," 2013 International Computer Science and Engineering Conference (ICSEC), Nakhonpathom, Thailand, 2013, pp. 31-36, doi: 10.1109/ICSEC.2013.6694748.
2. A Review on Server Based Code Editor Sonali R. Gujarkar, Samprada D.Nimrad, Shital Meshram INTERNATIONAL JOURNAL FOR RESEARCH IN EMERGING SCIENCE AND TECHNOLOGY, SPECIALISSUE-1-JAN-2017
3. Kurniawan, Aditya & Soesanto, Christine & Wijaya, Joe. (2015). CodeR: Real-time Code Editor Application for Collaborative Programming. *Procedia Computer Science*. 59. 510-519. 10.1016/j.procs.2015.07.531.
4. International Research Journal of Engineering and Technology (IRJET), BROWSER BASED CODE EDITOR. <https://www.irjet.net/archives/V8/i5/IRJET-V8I5440.pdf>
5. Ramdas Vankdothu, Dr. Mohd Abdul Hameed, Husnah Fatima "A Brain Tumor Identification and Classification Using Deep Learning based on CNN-LSTM Method" *Computers and Electrical Engineering*, 101 (2022) 107960
6. Ramdas Vankdothu, Mohd Abdul Hameed "Adaptive features selection and EDNN based brain image recognition on the internet of medical things", *Computers and Electrical Engineering*, 103 (2022) 108338.
7. Ramdas Vankdothu, Mohd Abdul Hameed, Ayesha Ameen, Raheem, Unnisa "Brain image identification and classification on Internet of Medical Things in healthcare system using support value based deep neural network" *Computers and Electrical Engineering*, 102 (2022) 108196.
8. Ramdas Vankdothu, Mohd Abdul Hameed "Brain tumor segmentation of MR images using SVM and fuzzy classifier in machine learning" *Measurement: Sensors Journal*, Volume 24, 2022, 100440.
9. Ramdas Vankdothu, Mohd Abdul Hameed "Brain tumor MRI images identification and classification based on the recurrent convolutional neural network" *Measurement: Sensors Journal*, Volume 24, 2022, 100412.
10. Bhukya Madhu, M.Venu Gopala Chari, Ramdas Vankdothu, Arun Kumar Silivery, Veerender Aerranagula

- ” Intrusion detection models for IOT networks via deep learning approaches ” Measurement: Sensors Journal, Volume 25, 2022, 100641
11. Mohd Thousif Ahemad ,Mohd Abdul Hameed, Ramdas Vankdothu” COVID-19 detection and classification for machine learning methods using human genomic data” Measurement: Sensors Journal, Volume 24, 2022, 100537
  12. S. Rakesh <sup>a</sup>, NagaratnaP. Hegde <sup>b</sup>, M. VenuGopalachari <sup>c</sup>, D. Jayaram <sup>c</sup>, Bhukya Madhu <sup>d</sup>, MohdAbdul Hameed <sup>a</sup>, Ramdas Vankdothu <sup>e</sup>, L.K. Suresh Kumar “Moving object detection using modified GMM based background subtraction” Measurement: Sensors ,Journal, Volume 30, 2023, 100898
  13. Ramdas Vankdothu,Dr.Mohd Abdul Hameed, Husnah Fatima “Efficient Detection of Brain Tumor Using Unsupervised Modified Deep Belief Network in Big Data” Journal of Adv Research in Dynamical & Control Systems, Vol. 12, 2020.
  14. Ramdas Vankdothu,Dr.Mohd Abdul Hameed, Husnah Fatima “Internet of Medical Things of Brain Image Recognition Algorithm and High Performance Computing by Convolutional Neural Network” International Journal of Advanced Science and Technology, Vol. 29, No. 6, (2020), pp. 2875 – 2881
  15. Ramdas Vankdothu,Dr.Mohd Abdul Hameed, Husnah Fatima “Convolutional Neural Network-Based Brain Image Recognition Algorithm And High-Performance Computing”, Journal Of Critical Reviews, Vol 7, Issue 08, 2020(Scopus Indexed)
  16. Ramdas Vankdothu, Dr.Mohd Abdul Hameed “A Security Applicable with Deep Learning Algorithm for Big Data Analysis”, Test Engineering & Management Journal, January-February 2020
  17. Ramdas Vankdothu, G. Shyama Chandra Prasad “ A Study on Privacy Applicable Deep Learning Schemes for Big Data” Complexity International Journal, Volume 23, Issue 2, July-August 2019
  18. Ramdas Vankdothu, Dr.Mohd Abdul Hameed, Husnah Fatima “ Brain Image Recognition using Internet of Medical Things based Support Value based Adaptive Deep Neural Network” The International journal of analytical and experimental modal analysis, Volume XII, Issue IV, April/2020
  19. Ramdas Vankdothu,Dr.Mohd Abdul Hameed, Husnah Fatima” Adaptive Features Selection and EDNN based Brain Image Recognition In Internet Of Medical Things “ Journal of Engineering Sciences, Vol 11, Issue 4 , April/ 2020(UGC Care Journal)
  20. Ramdas Vankdothu, Dr.Mohd Abdul Hameed “ Implementation of a Privacy based Deep Learning Algorithm for Big Data Analytics”, Complexity International Journal , Volume 24, Issue 01, Jan 2020
  21. Ramdas Vankdothu, G. Shyama Chandra Prasad” A Survey On Big Data Analytics: Challenges, Open Research Issues and Tools” International Journal For Innovative Engineering and Management Research, Vol 08 Issue08, Aug 2019.
  22. Vankdothu, R., Hameed, M.A. “An Effective Congestion and Interference Secure Routing Protocol for Internet of Things Applications in Wireless Sensor Network “ Wireless Personal Communication Journal 140, 143–161 (2025)

23. Vankdothu, R., Bhukya, H. & Bhukya, R.R. “Hybrid TDR-MI Based Wireless Sensor Network for Underground Water Pipeline Leakage Detection and Localization Using Pressure Residuals and Classifiers Wireless Personal Communications 139, 803–823 (2024).
24. Vankdothu, R., Cheng, X. “Energy Efficient TDMA and Secure Based MAC Protocol for WSN Using AQL Coding and ASGWI Clustering”. Wireless Personal Communications 136, 2125–2143 (2024)
25. Vankdothu, R., Hameed, M.A., Fatima, H. *et al.* Multicast Scaling in Heterogeneous Wireless Sensor Networks for Security and Time Efficiency. Wireless Personal Communications (2025).
26. Vankdothu, R., Hameed, M.A., Fatima, H. *et al.* Multicast Scaling in Heterogeneous Wireless Sensor Networks for Security and Time Efficiency. Wireless Personal Communications (2025)
27. Ramdas Vankdothu, Mohd Abdul Hameed” Brain MRI Images for Tumor Detection using Storage Optimization Technique”, Mobile Radio Communications and 5G Networks, Lecture Notes in Networks and Systems, 425-437, Springer .
28. Bandi Krishna , Ramdas Vankdothu , Varun Revuri and B. Prashanth” A brain tumor identification using convolution neural network in the deep learning” MATEC Web of Conferences 392, 01131 (2024) ,<https://doi.org/10.1051/mateconf/202439201131> ICMED 2024

## BIBLIOGRAPHY



I am Sadiya Mazarath from the Department of Computer Science and Engineering. Currently, pursuing 3<sup>rd</sup> year at Balaji Institute of Technology and Science. My research is done based on “Online Code Editor”.



I am V.Rajeshwari from the Department of Computer Science and Engineering. Currently, pursuing 3<sup>rd</sup> year at Balaji Institute of Technology and Science. My research is done based on “Online Code Editor”.



I am T.Bhargavi from the Department of Computer Science and Engineering. Currently, pursuing 3<sup>rd</sup> year at Balaji Institute of Technology and Science. My research is done based on “Online Code Editor”.



I am Syed Raiyan from the Department of Computer Science and Engineering. Currently, pursuing 3<sup>rd</sup> year at Balaji Institute of Technology and Science. My research is done based on “Online Code Editor”.