

Code Sandbox -An Interactive Online Code Compiler

Nikhil Kumar, Vaibhav Tomar
School of Engineering and Technology
Department of Computer Science and Application,
Sharda University, Greater Noida (UP) India
Nikpanwar9@gmail.com

Abstract: To overcome the shortcomings of the current online coding platforms, this article suggests "Project Code Sandbox," an interactive online code compiler. The inability of current methods to save data in the cloud impairs user experience by limiting code persistence and cross-device accessibility. Project Code Sandbox uses Amazon's EC2 service to provide cloud storage for user code to remedy this. This increases workflow flexibility and promotes collaboration by allowing users to change and view their code at any time and from any location. To provide an online coding environment that is more user-centric, the paper introduces the idea behind Project Code Sandbox and describes its intended implementation utilizing EC2.

Keywords: Online compiler, Interactive coding, Cloud IDE, Real-time collaboration, Web Development, developer community

1. Introduction:

While coding is made easier by the availability of online code editors and compilers, cloud storage remains a persistent difficulty. This makes it more difficult for users to store and retrieve their code from anywhere. In order to overcome this constraint, the present study suggests "Project Code Sandbox," an interactive online code compiler that incorporates cloud storage using the EC2 service from Amazon Web Services. With this functionality, users can:

- **Code can be kept in the cloud:** With Project Code Sandbox, users may store their code on a remote server that is accessible from any device, doing away with the requirement for local storage.
- **Modify code remotely:** Cloud storage encourages flexibility and teamwork by enabling users to view and edit their code from any location with an internet connection.

The integration of cloud storage functionality is the main subject of this study, which investigates the conception and execution of Project Code Sandbox. The use of EC2 for persistent code storage will be described in depth, and the advantages for programmers looking for a more approachable and cooperative writing environment will be discussed.

2. Literature Review

Numerous online code editors and compilers now in use include functionality that are pertinent to your project. But one significant drawback you point out is the absence of cloud storage for user code.

Cloud Storage's Advantages for Code:

Accessibility: Cloud storage fosters remote development and collaboration by enabling users to view and modify their code from any internet-connected device.

Version Control: Version control is made easier by cloud storage, which lets users keep track of changes, roll back to earlier iterations, and work together efficiently.

Scalability: The capacity to store larger volumes of code without being constrained by local storage is made possible by cloud storage.

Cloud Storage with Amazon EC2

It seems good that you want to use Amazon's EC2 service for cloud storage. Scalable and secure virtual servers with the ability to store user code are offered by EC2.

Project Code Sandbox's Differentiation:

The inclusion of cloud storage is a significant feature that sets this project apart from other online code editors and compilers. Project Code Sandbox gives customers the ability to put their code directly on the cloud, in contrast to existing alternatives, enabling:

Access Anytime, anywhere: This feature allows users to work on their code from any internet-connected device, increasing flexibility and enabling on-the-go development.

Better Collaboration: Several users can work on the same project at once thanks to cloud storage's smooth collaboration features.

Improved Project Management: Version control and project organization are made easier for users by allowing them to have a central repository of their code.

Future Research Directions:

This study suggests using the EC2 service from Amazon Web Services (AWS) to develop a permanent storage solution for user code.

There are various ways in which future research directions can build upon this foundation:

Security and Access Control: To safeguard user code kept on the cloud server, put strong security measures in place. This covers encryption of sensitive data, access control methods, and user authentication.

Integration of Version Control: Integrate with a version control system such as Git, which enables users to monitor changes made to the code, roll back to earlier iterations, and work together efficiently on projects.

AI-Powered Assistance: Include AI features to advise code completion, spot possible mistakes, and suggest improvements. This can speed up development and improve the user experience.

3. Methodology

Project Code Sandbox: Development Methodology

The goal of Project Code Sandbox is to offer an effective and user-friendly online code compilation tool. This is a summary of our development process:

Selection of Technology Stack:

- **Frontend:** To guarantee a fluid and responsive user experience, we'll make use of a contemporary JavaScript framework like React or Vue.js.
- **Backend:** The compilation engine will be powered by a server-side language, such as Node.js, which will communicate with language-specific compilers and handle user requests.
- **Cloud Infrastructure:** For scalability and effective resource allocation, we'll make use of cloud services (such as AWS and Google Cloud Platform). As a result, we can manage several users and offer quick compilation times.

The Process of Development:

- **Modular Design:** The project will be divided into more manageable, smaller components (such as an editor, compiler, and features for collaboration). This encourages easier maintenance and code reuse.
- **Agile Development:** An iterative development methodology will be used. This entails completing tasks in sprints, routinely obtaining user input, and iteratively enhancing the platform in response to user requirements.
- **Version Control:** Git will be used to track changes for future reference, control code versions, and enable collaboration.
- **Testing:** To ensure that various modules operate together flawlessly, integration tests will be used in addition to unit tests for core functions.

Important Elements:

- **Language Support:** Depending on user demand, progressively add more languages after starting with well-known ones like HTML, CSS, Python, JavaScript, and so on.
- **Real-time Code Editing:** Code should be written and edited by users directly in the browser, with the output window updating instantly.
- **Code Compilation and Execution:** To easily build and run code snippets, integrate language-specific compilers.
- **Sandbox Environment:** Separate user projects to avoid disputes and guarantee a secure setting for testing.
- **Collaboration Tools:** To facilitate collaborative coding, incorporate tools like chat capabilities, real-time code sharing, and shared workspaces.
- **Persistent Storage:** Provide users with the option to version their projects and preserve them for later use.
- **Learning Resources:** Include documentation, code samples, and tutorials to help users of all skill levels.

Security:

- **Secure Code Practices:** Adhere to secure coding practices to minimize vulnerabilities that could compromise user data or the platform itself.
- **Data Encryption:** Implement data encryption for sensitive user information (e.g., usernames, passwords) both in transit and at rest.
- **Access Control:** Establish a user authentication system with appropriate access levels to ensure project privacy.

Implementation and Upkeep:

Establish a Continuous Integration/Continuous Delivery (CI/CD) pipeline to automate code deployment, testing, and upgrades, thereby guaranteeing a seamless and effective release process.

Monitoring and Logging: Keep an eye on user behavior and application performance to spot possible problems early on and take proactive measures to fix them.

Scalability: Create an infrastructure that can grow with a user base and handle more traffic.

This methodology provides a framework for developing a robust and user-friendly online code compiler. Regular user feedback and feature iteration will be crucial for Project Code Sandbox to evolve and become a valuable tool for the developer community.

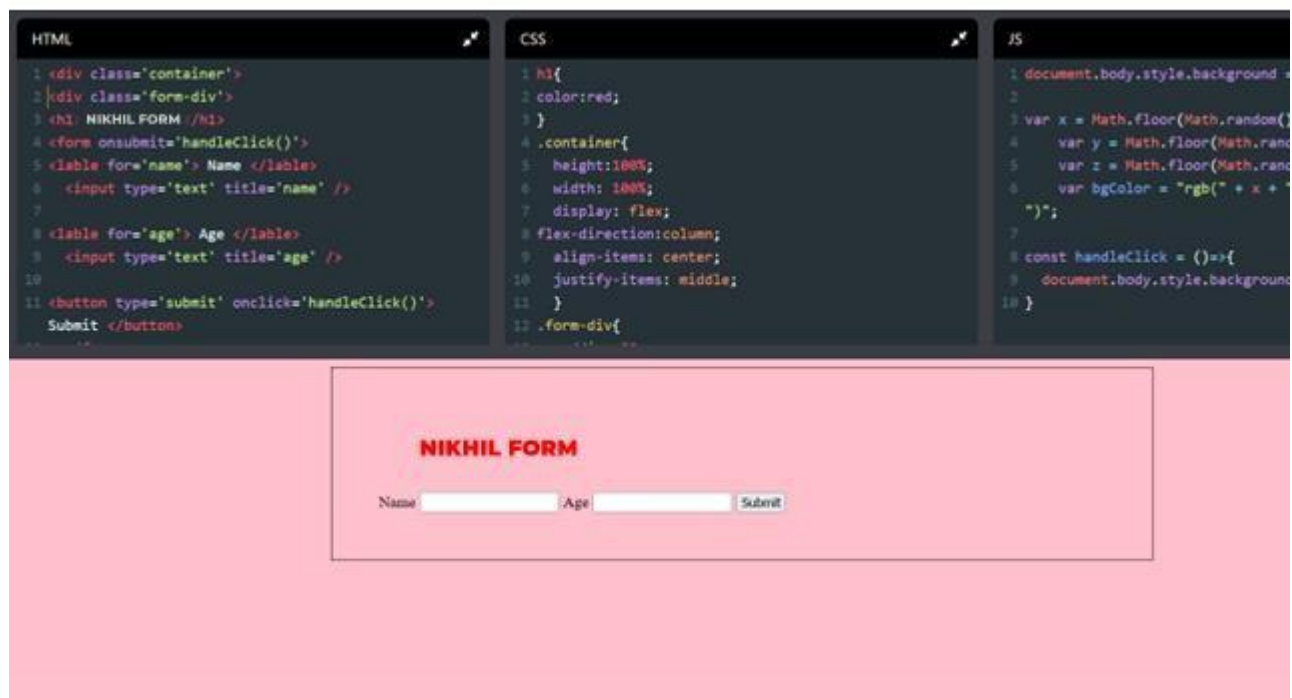


Fig 1. Flow Chart of System

4. Conclusion

In conclusion, this paper presented Project Code Sandbox, an interactive online code compiler that addresses the limitations of current online coding platforms. Project Code Sandbox offers a user-friendly interface, real-time compilation and error correction, and support for multiple programming languages.

By leveraging cloud storage through services like AWS EC2, future iterations of Project Code Sandbox will enable users to seamlessly store, access, and edit their code from any location.

This innovative feature will empower programmers to collaborate effectively and enhance their coding experience. Overall, Project Code Sandbox has the potential to become a valuable tool for both novice and experienced programmers alike.

5. References

- <https://reactjs.org/docs>
- <https://codemirror.net/2/>
- <https://rapidapi.com/judge0-official/api/judge0-ce/>
- <https://uiwjs.github.io/react-codemirror/>
- <https://stackoverflow.com/search?q=react>