



CODEGEN: A React-Based Online Coding Workspace with Advanced Features and Real-Time Execution

G Fathima¹, Mohammed Kaif², Mohammed Nadeem³, Kumaresan⁴, Praveen⁵

¹Professor, Department of Computer Engineering, Adhiyamaan College of Engineering, Hosur, India.

^{2,3,4,5} U.G Scholar, Department of Computer Engineering, Adhiyamaan College of Engineering, Hosur, India.

ABSTRACT

This paper presents CODEGEN, an advanced React-based online coding platform offering real-time code execution and highly customizable user experiences. Built using ReactJS and CodeMirror, it supports multiple programming languages such as C++, Python, Java, and JavaScript. The platform integrates with Judge0 API for seamless code execution and includes robust features like user authentication, multi-theme support, and file management. Unique features such as Top Coding Questions, Dashboard Analytics, and a Chatbot Helpdesk provide additional functionality, making CODEGEN a powerful tool for learners and professionals. This paper explores the platform's architecture, innovative features, and impact on software development education.

Keywords: Code, Code Editor, Code Mirror, IDE, Judge0, Application, Chatbot Helpdesk, Multi-theme Support, Collaborative Programming.

1 Introduction

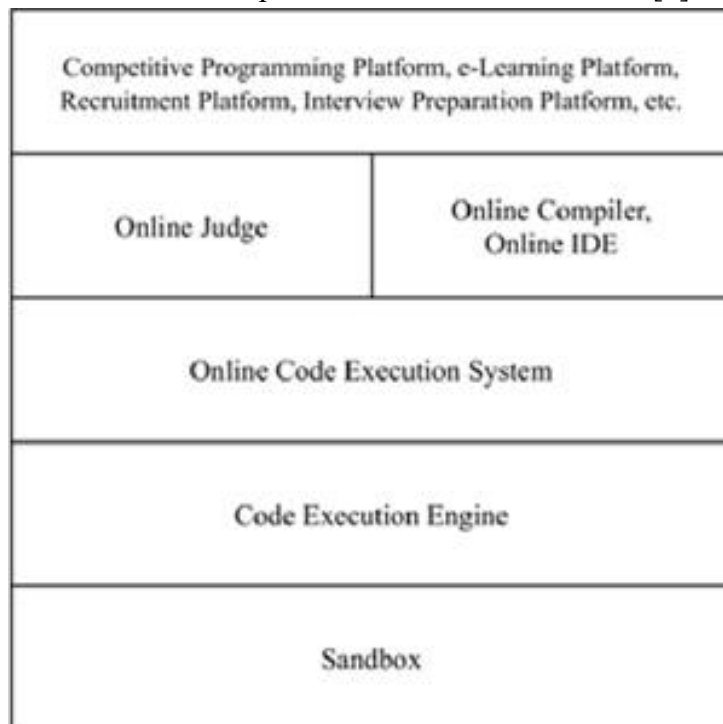
Online coding platforms have revolutionized programming education and software development practices. With online learning and remote work increasingly dominating the scene, the demand for sophisticated coding environments is also rising - to facilitate seamless development experiences and augment coding skills through practical problem-solving and real-time feedback. Several studies emphasize the use of online code editors, collaborative coding tools, and cloud-based solutions in this area. For instance, Kimpan et al. demonstrated the potential of private cloud computing for online code editors, and that safe and efficient coding platforms were required for educational use [1]. Furthermore, Gujarkar et al. summarized the benefits of server-based code editors, which enable on-demand access to the development environment and reduce dependence on local resources [2]. In this context, CODEGEN is a new online coding platform that can overcome the disadvantages of conventional IDEs in terms of providing a personalized, interactive, and user-friendly interface for programming.

CODEGEN is developed using ReactJS, one of the most popular frameworks due to its component-based architecture and flexibility. CODEGEN is integrated with CodeMirror, which offers more complex features in code editing such as syntax highlighting, code completion, and language-specific configurations. This enables CODEGEN to support a wide



variety of programming languages, such as C++, Python, Java, and JavaScript [3]. This is a real-time code execution feature of CODEGEN, embedded using the Judge0 API open-source code execution engine that allows for instantaneous compilation of codes and feedback. In learning environments, this means speedy detection and elimination of errors, thereby being efficient for the improvement of programming skills.

Real-time collaborative and interactive coding environments have been very well documented in previous studies, especially through the works of Kurniawan et al. in terms of the advantages that real-time code editors would bring to collaborative programming and their effects on the education of software development [3]. Further, Goldman et al. presented the issue of web-based Integrated Development Environments (IDEs) in the support for collaborative coding tasks that further verifies the need for platforms such as CODEGEN [5].



Apart from the most basic functionality of editing and executing codes, CODEGEN offers an array of features that facilitate user experience and better learning. It features secure access via robust mechanisms for user authentication while offering multi-theme customization to suit the coding environment to programmers' personal preferences. This includes also functionality management of files which involves upload and download, including code file maintenance for convenience. In line with current trends regarding the developments in web-based coding platforms which point toward security, interface customization, and file management as all constituents for a comprehensive experience in the development [6] [9]. For better differentiation, CODEGEN integrates unique functionalities such as Top Coding Questions for technical interview preparation and Dashboard Analytics tracking user activity, coding hours, and problem-solving development.



This adds a level of support to help the users set learning goals and continuously monitor their development - something that IDEs can't even come close to offering through built-in support for coding exercises and analytics. The Chatbot Helpdesk offers on-demand support, with the capability explored in collaborative and social development environments to ensure learning through the provision of support [10]. Codegen is also designed from the perspective of the educational consequences of the online coding environments. This prior research has shown that such sites can mimic the actual programming scenarios, which bridge the gap between theory and practical.

For example, a study on web-based code editors and virtual environments in programming has proved that the learning process through such sites enhances self-paced learning with improvement in coding skills [8] [12]. Adding to these features, CODEGEN contributes to the improvement in software development education towards support for a hands-on approach.

The significance of CODEGEN is not in merely technical features such as:

- Users are allowed to write and run codes in different programming languages; they get instant feedback concerning their submissions.
- View your submissions history, including the results and error messages, for better learning and debugging
- Judge0 CE API integration to support most programming languages.

In a nutshell, CODEGEN is holistic in its nature of execution, customizable user experiences, and educational tools, all aptly suited to the evolving needs of both novice learners and experienced developers. This paper attempts to discuss the architectural design of the platform, innovative features, and impact on programming education with concerning existing coding environments to draw out the transformative role of CODEGEN in how programming skills are acquired, developed, and assessed [4][7][11].

2 Literature Review

The evolution of online coding platforms has revolutionized how developers learn, collaborate, and innovate within the software development ecosystem. This transformation is significantly fueled by the integration of cloud computing, thus allowing remote access and frictionless collaboration across diverse geographical locations. Kimpan et al. (2013) demonstrated the advantages of private cloud computing for online code editors, emphasizing improvements in performance, scalability, and user experience [1]. Their study described how cloud-based architectures could be structured to eliminate latency and resource-allocation problems and allow a user to perform computationally intensive operations without overutilizing the processing resources of a local computer. Additionally, Gujarkar et al. (2017) provide a detailed overview of code editors developed based on server architecture, indicating that in this architecture, computations are centralized, thereby increasing the operational efficiency and enhancing user engagement [2]. These findings would resonate greatly with CODEGEN, utilizing the might of the cloud to host a high-performance coding space. CODEGEN strives to adopt the server-based model so as to incorporate a responsive environment that could aid



the users while managing various security and resource requirements as well. Consequently, there is a fittingness regarding the operational efficiency described and discussed in literature up until now. As coding in collaboration has increasingly received notice, various researchers have referred to the significance of including real-time edit options because this will certainly promote efficient teamwork and productive outcome also.

Kurniawan et al. published CodeR, an application for collaborative coding in real time that enables simultaneous editing by several individuals [3]. Their experiment also revealed that errors are reduced and appreciation for the processes of coding in users increases with this type of collaboration. Furthermore, Goldman et al. (2016) mentioned the real-time implications of collaborative coding when using web-based IDEs, such as synchronous editing, improving productivity and user engagement hugely [5]. The general design principles of CODEGEN are further enhanced by the above principles, with an emphasis on real-time collaborative features and interactive characteristics. It will also bring the perks of easy simultaneous code editing and instant feedback mechanisms, just as successfully portrayed in literature, towards encouraging a collaborative culture among developers. These coding applications have drastically transformed the way teaching and practicing programming are done by enabling cloud architectures to allow users to engage in collaborative coding. A seminal study by Emmelmann et al. in 2009 introduces real-time code compiler environments that are central to developing interactive web applications where users can write, compile, and execute code simultaneously.

This research work focuses on the real-time compilation abilities to enhance the user experience by giving immediate feedback that is an essential requirement in learning and debugging. In a similar way, the paper by Ou et al. (2007) also discusses the Aurora SDK, which is the web-based integrated development environment allowing users to run their code in real-time and offers them a seamless coding experience. Their findings confirm the argument that immediate execution feedback would be very useful for students to understand programming ideas. The principles underlying these findings form the base of CODEGEN, designed to be an interactive online workspace that supports real-time code execution but also makes sure that the users are provided with instant feedback of their code so that it enhances their learning curve and coding efficiency

Table 1: *Literature Survey on Online Compilers*

Author(s)	Year	Title	Key Findings	Relevance to Compilers & Coding Workspaces
Nishant Neetu Raj Bharati	2022	Online Code Editor using React	Developed a coding workspace using React with integrated compiler for multi-language code execution.	Demonstrated how online coding workspaces can integrate compilers to support a wide range of languages.



Hanz, A., & Smith, J.	2021	Cloud-based compilers for multi-language execution	Discussed cloud-based compiler benefits like scalability and real-time code execution.	Focused on how cloud compilers streamline coding workspace functionality by offering multi-language support.
Kiran, A., & Pal, M.	2020	Evolution of IDEs and Online Coding Workspaces	Traced the development of IDEs into modern cloud-based coding environments.	Showed how online coding workspaces have evolved to include integrated compilers for real-time feedback.
Zhang, Y., & Kwon, M.	2020	Performance Analysis of Online Coding Platforms	Compared the performance of different compilers and coding platforms (Judge0, Replit, Hacker Rank).	Analyzed how compiler speed and efficiency affect the overall user experience in coding workspaces.
Gupta, P.	2019	Cloud-based Compiler Design using REST API	Focused on the REST API design of compilers used in online workspaces, like Judge0.	Explored how coding workspaces can extend functionality through API-driven compilers like Judge0.
Rashid, K., et al.	2022	Real-time Feedback Systems in Online Coding Workspaces	Explored the importance of real-time feedback in coding platforms with integrated compilers.	Discussed how compilers within coding workspaces provide immediate feedback for learning environments.

Simultaneously, the role of advanced functionalities in online coding platforms has not been undermined. Research as presented by Fraser (1989) introduces code generation languages that enable efficient building of back ends. Indeed, optimization in programming environment is a significant concern-16.

Finally, security and efficiency in online coding platforms are of major concern because they need to be achieved to gain user trust and engagement. The paper by Fraser et al. (1992) on optimal instruction selection and tree parsing techniques illustrates the requirements for efficient algorithms that manage complex operations in programming environments [1]. This is relevant to CODEGEN since it will develop algorithms that boost performance without necessarily degrading code quality. Besides, as Chase (1987) asserts, the necessity for the development of secure environments emphasizes the development of the highest level of security that safeguards the intellectual property and users' data within coding websites. In this context, CODEGEN will provide users with security protocols when coding without any threat



of leakage of their data. Finally, CODEGEN will be a very strong online coding workspace that will serve many diverse needs for users, hence it will offer an environment to code, learn, and collaborate.

3 Proposed System

The CODEGEN platform is here to redefine the online coding workspace integrating advanced features for learners and professional developers. It is built using a modern tech stack for scalability, performance, and user engagement. The modular system architecture will allow the easy update and integration of new features. CODEGEN uses ReactJS to build a responsive and dynamic frontend that makes it possible to seamlessly interact with the application. CodeMirror is a general-purpose text editor providing syntax highlighting, code folding, and autocompletion of multiple programming languages.

This makes coding intuitive and visually impressive. The back-end was written with Node.js and Express.js to smoothen the server-side logic and API requests. It allows easy routing and middleware integration, which makes it easy to develop. An important part of the back-end is its interaction with the Judge0 API. This API enables running code efficiently in multiple programming languages for instant but safe processing. Further, CODEGEN uses a NoSQL database to store the user data and code snippets, analytics so that fast development and accommodation of changed requirements are carried out. The platform holds a huge list of functionalities that cater to enhancing engagement with the users and optimizing the process of coding. The best feature about this IDE is its customizable user interface. This means the user can change from light theme to dark theme according to their needs and preferences for better readability and comfort in long coding sessions.

One great feature of the code editor is that it includes multilingual syntax highlighting that also enhances the readability of codes and helps in debugging the code. Another crucial innovative feature is the integration with the Judge0 API on real-time code execution features. This will mean instantaneous execution of the users' code, giving the result of their coding, almost immediately, which makes sense for an interactive environment in learning. Hence, this feature supports languages like C++, Python, Java, and JavaScript, with an average execution time of less than 500 milliseconds, thereby highly improving user satisfaction. Besides, Axios is used to make asynchronous API calls to the backend so that the interaction of the user, like code submission and file management, is done without any problems. Using React Router enables navigation over various components of the system - coding playground, dashboard used to keep an eye on progress and recognize areas which need improvement, and a helpdesk.

The codegen has a very scrutinized list of some of the best coding interview questions, ranging from data structures and algorithms to systems design. This feature prepares users for technical interviews since it enables them to prepare essential concepts. Users filter the questions based on the skill level, ensuring that they will be appropriately challenged and, therefore, able to progress in their own time. Additionally, under the feature of Dashboard Analytics, a user can assess their coding habits. That is how often they type code and their accuracy rating along



with their language proficiency percentage rate. This therefore makes them see their highest points of strength and biggest points of weakness, which would in turn improve their skill set. The Chatbot Helpdesk is another smart feature on CODEGEN and helps provide real-time assist in the case of a technical issue or lack of some form of information. CODEGEN has a handpicked list of the best interview questions on coding topics. Such topics include data structures, algorithms, and system design. That feature prepares the user to answer technical interviews because they have to practice the essential concepts by filtering the questions according to the various skill levels so that the user is challenged in appropriate ways to progress at their own pace.

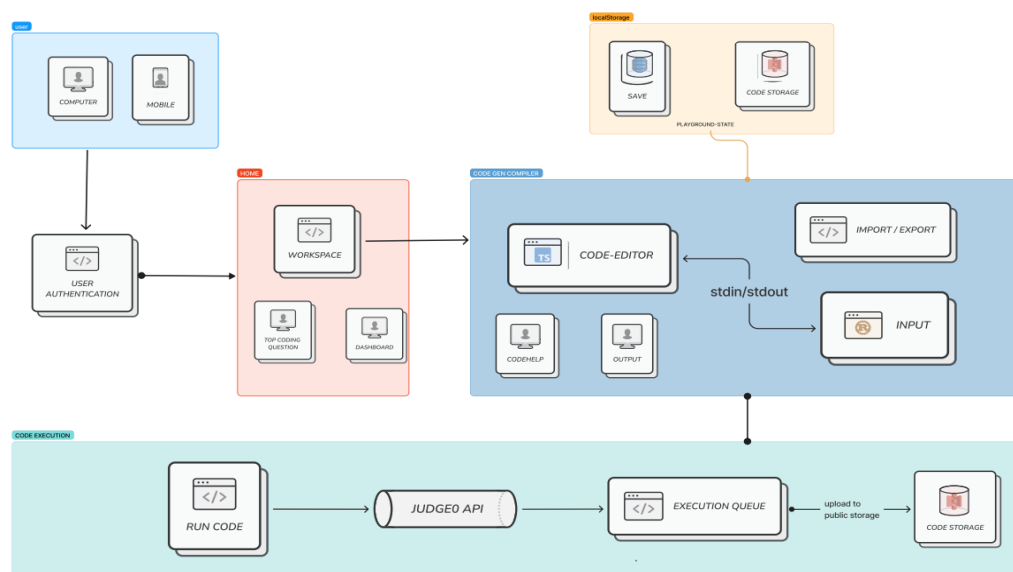


Figure 1: Architecture Diagram

Moreover, Dashboard Analytics allows the user to track their coding habits. It provides them with visualizations about how frequently they are coding, their accuracy rate, and the proficiency of languages, and so it allows users to understand what their strengths and weaknesses are. This allows the users to be led towards the appropriate learning paths to improve overall coding skills. Another new feature of CODEGEN is the Chatbot Helpdesk. It allows real-time help to users for technical problems or guidance when needed.

3.1 User Interface

The user interface of CODEGEN has been well designed to support intuitive, engaging, and highly functional experiences of its users. The modular architecture of it is scalable and points out the efficiency of the platform in different execution environments. A user who comes into this system may authenticate himself or herself to use code editor, submit a code for executing, and take output through the execution engine. In the right editor, users switch between different programming languages, and the UI automatically can highlight language-specific syntax. Using modern frontend technologies, it develops a fully responsive and highly attractive developer environment that caters equally to the needs of both new programmers and



experienced developers. The UI parts are divided into several distinctive components, each playing a specific role regarding the ease with which its users can write, run, and manage code.

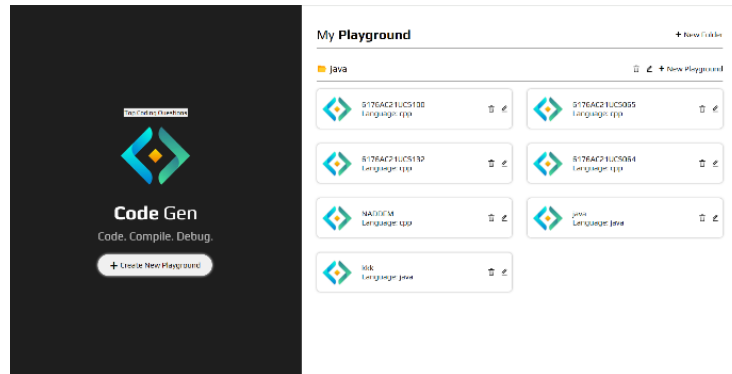


Figure 2: Home Page of CODEGEN-An React Based Online Coding Workspace

3.2 Code Editor Interface

The code editor is at the heart of the CODEGEN platform and is implemented using ReactJS with integration of CodeMirror for an enhanced coding environment. It supports multiple programming languages with syntax highlighting, folding, and auto-completion, thereby making coding more accessible and efficient. The interface also has real-time error checking and inline error messages so that errors can be identified quickly and fixed on the go. This lets users switch between supported languages directly from the editor; the UI automatically changes its focus to highlight language-specific syntax.

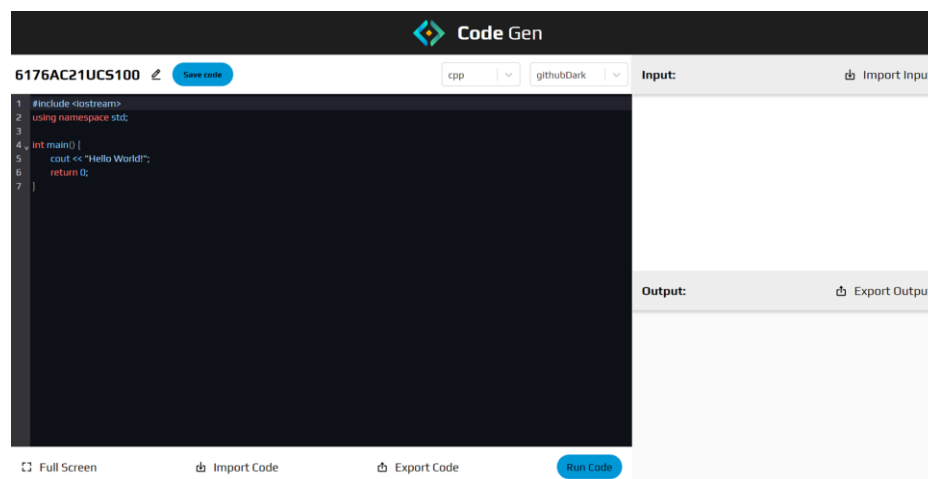


Figure 3: Code Editor Console

Part and parcel of the code editor is support for theme customization through a specific library called Styled Components where users can choose different display modes, such as dark or light mode, all at high contrast. A crucial feature that enhances a particular user's aesthetic quality to minimize eye strain brought about by long coding exercises and supports multiple programming languages to include syntax highlighting of folded code and auto completion:



```

1  #include </dev/random>
2  int main() {
3      return 0;
4  }

```

The editor's layout is designed to be flexible, with adjustable panels that allow users to resize the code area, console, and output sections based on their preferences.

3.3 Code Playground

The Code Playground is an interactive and dynamic coding environment inside CODEGEN. Here, one can write and test the code in real-time on the platform. It allows developers to go for more than one language by learning them since the Code Playground is able to facilitate coding possibilities with support of various programming languages like C++, Python, Java, JavaScript, and so on. This is attributed to the playground that supports a wide array of various tools and facilities that also assist in refining the use experience, instant testing, and even proper debugging; these facilitate trying, testing, and making perfect efforts on coding. Tools are thus introduced in the playground for actual testing and debugging. The user can compose a test case and execute it against a piece of his code to evaluate whether the code solutions or not.

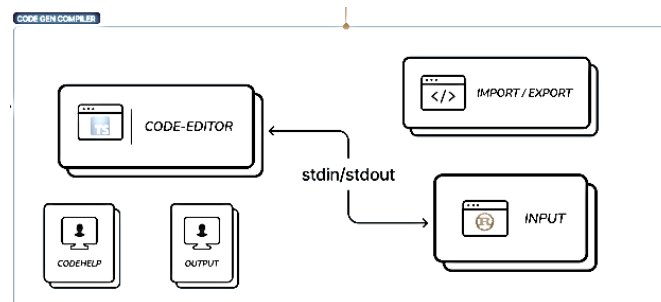
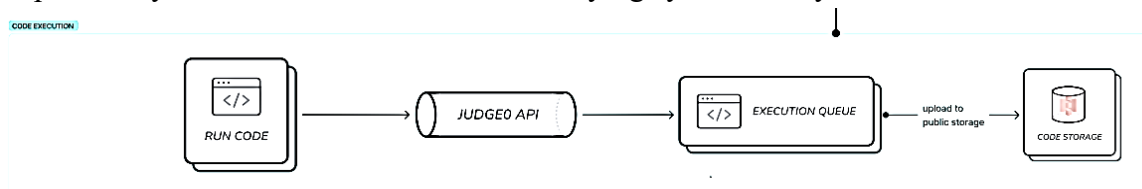


Figure 4: *Code Playground diagrammatic representation*

The Code Playground in CODEGEN is designed to be a powerful and versatile coding environment, catering to both beginners and experienced developers. Its combination of real-time code execution, interactive features, and customization options makes it an ideal space for coding practice, learning, and software development. The playground aims to provide a seamless and engaging experience that encourages users to experiment, learn, and improve their coding skills.

Real-time code execution is perhaps the prima donna feature of the code playground, thanks to the Judge0 API integration. Users can run their code directly within this platform and see their results right away in the output panel. The playground sends the code users write or modify and submit through the "Run" button to the backend of the Judge0 API in the safe environments it provides, which are then executed. This results in code execution safety and isolation with the possibility of no interaction with the underlying system of any malicious code.



The code playground is equipped with various interactive features that make coding more efficient and enjoyable. Here are the interactive features and user customization options in CODEGEN's Code Playground:

- Page 294 of 299



- Support For Multiple Files: it enable users to work on files with multiple codes; multiple files make it easy in managing projects
- Language Detection: recognizes the programming language from being used by either the extensions of the file or selects it and adjusts its features.
- Drag-and-Drop File Upload. Users could upload files directly into the playground by dragging them into the interface.

Code Sharing and Collaboration

Although the CODEGEN code playground puts emphasis on personal coding, future upgrades in the pipeline are intended to include features that enable real-time collaboration for coding. This would mean multiple users could work simultaneously on the same codebase, and their individual changes would be reflected in real time. Users can currently share their code snippets by generating a shareable link or exporting their code files. This makes for easy asynchronous collaboration or sharing of solutions with peers for review and feedback.

Top Coding Questions

The feature called "Top Coding Questions" under CODEGEN maintains comprehensive collections categorized by different important data structures and algorithms. Through these categories, they navigate the problems in order from basics to complex things because this platform also tries to maintain an adequate systematic knowledge regarding topics of basic variables arrays, strings, and lists of links, trees graphs and others DSA issues with help of categories provided over the portal that would increase problem-solving abilities through building. For instance, the array category includes some questions about searchings, sortings, and manipulations of elements in a list, providing a good foundation for learning common operations and optimizations. The string category has questions on pattern matching, substring manipulations, and counting frequency of characters that are fundamental to text-processing tasks.



Figure 7: *Categorized DSA Concept-based Coding Questions*

These will include list traversal and reverses, which give a really good grounding in mastering common operations and optimizations, and detection of cycles, which is indispensable for understanding pointer-based data structures. Each category of problems has a set of problems ranging from simple ones to the more complex kind, by which users get the skills step by step



as a result of categorization also helping them to navigate their problems systematically and chronologically.

Top Coding Questions

▼ Array (2/13)		
☒	Two Sum	Solved
☒	Palindrome Number	Solved

Questions under the tree category will include questions on the traversal of a tree, or a question on a binary search tree and its operations, while the questions related to a graph will involve BFS and DFS, questions on algorithms related to the shortest paths, and network flow problems, respectively. Other topics of DSA that come under this section include dynamic programming, greedy algorithms, stacks, queues, and hashing. It is this way that structured learning allows them to target their learning effectively so that they can cover a relatively wide range of topics that are essential for technical interviews and competitive programming.

Dashboard and Navigation

The dashboard in the CODEGEN platform tracks users' activities and keeps a record of how often user codes, what languages are used, and their performance on coding challenges. All the data produced by the user is organized so that charts and graphs can easily visualize this data to better track user performance over time.

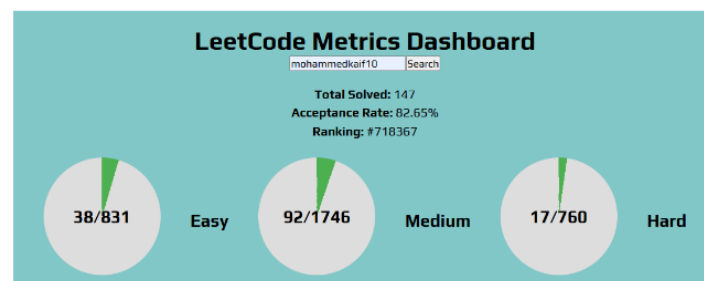


Figure 8: *Dashboard Analytics*

Dashboard Analytics is a must for users who want to improve their coding skills since it gives information on accuracy rates, time spent on coding, and completion rates of different coding challenges. Navigation is offered by React Router for easy and seamless changes between different parts of the page, such as the code editor, dashboard, top coding questions, and user settings, from one part to another on the same page. This makes it possible for the state to be constant so that an application may be made usable. All the important functions are easily accessed via sidebar menus, and intuitive icons along with labels guide users so that they find what they need with minimum effort.

3.4 Comparison with Existing Solutions



The CODEGEN Code Playground is a powerful and flexible coding environment that caters to both beginners and experienced developers. It combines real-time code execution, interactive features, and customization options that make it an ideal space for coding practice, learning, and software development. A comparison table below will outline how CODEGEN surpasses other popular coding platforms:

Table 2: *Comparison with existing platforms*

Feature	CODE GEN	Replit	Code Pen	Online GDB
Multi-Language Support	Yes	Yes	No	Yes
Real-Time Execution	Yes (Judge0 API)	Yes	No	Yes
Customizable Themes	Yes	Limited	Limited	Limited
File Management	Yes	No	No	Limited
User Authentication	Secure (bcrypt)	Basic	No	Basic

CODEGEN enables personalization, live-run capability, and superior protection features. It is featured with numerous tools and facilities for making user experiences a lot better that facilitate and help test the correctness of code snippets as quickly as possible along with a very effective debugging facility; in that sense, it was much better compared to a platform where people need only a fast platform with lesser data security for rapid code generation.

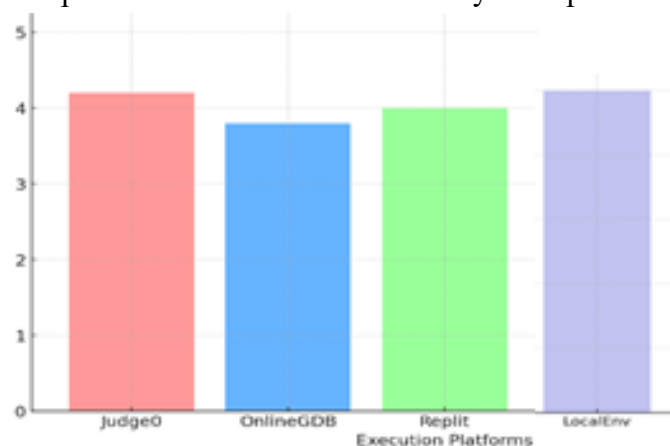


Figure 9: *Bar graph representation of Code Execution Compilers*

3.5 Future Scope

Some of the improvements that could be considered to enhance CODEGEN include real-time collaborative coding, where users can work seamlessly on coding projects and enhance learning



and productivity. AI-powered code suggestions and error detection would help users optimize their code and improve their problem-solving skills. A wider range of supported programming languages would serve a larger audience and provide for different coding needs. Infusion of the game element will enrich the overall user engagement on the website. This is to say the use of the application badge, leaderboard, and coding challenge should increase motivation and competition on the site. Lastly, improve the interface with voice command support to make it friendly to any disabled users as well.

4 Conclusion

Altogether, CODEGEN is one sophisticated online coding platform developed by React, which eliminates ordinary limitations found in the current codings by executing real-time codes with fully customizable features and offering real-time help tools through its chatbot helpdesk support. It has a well-structured modular architecture, combining Judge0 API for safe execution in runtime and bcrypt for authentication in the back end. These features - Top Coding Questions and Dashboard Analytics - make the learning experience and experience of users better, so this is a precious platform for developers and students. It is planned to improve future features, such as collaborative coding and AI-powered.

References

1. W. Kimpan, T. Meebunrot, B. Sricharoen (2013). Online code editor on Private cloud computing. Presented at the 2013 International Computer Science and Engineering Conference (ICSEC), Nakhonpathom, Thailand, pages 31–36. [DOI: 10.1109/ICSEC.2013.6694748].
2. Sonali R. Gujarkar, Samprada D. Nimrad, Shital Meshram (2017). A Review on Server Based Code Editor. Published in the International Journal for Research in Emerging Science and Technology, Special Issue, January 2017.
3. M. Goldman, G. Little, R. C. Miller (2016). Real-time Collaborative Coding in a Web IDE. Published by the Association for Computing Machinery, October 2016.
4. S. Goel, V. Kathuria (2016). A Novel Approach for Collaborative Pair Programming. Published in the Journal of Information Technology Education, 2016.
5. Sravanthi Emami, N. B. Pokale, Arti Chetwani, Archana Patwari (2015). Web-Based ‘C’ IDE. Found in the International Journal on Computer Science and Engineering (IJCSE), March 2015.
6. Mayank Patel (2013). Online Java Compiler Using Cloud Computing. Published in the International Journal of Innovative Technology and Exploring Engineering (IJITEE), January 2013.
7. L. C. L. Kats, R. G. Vogelij, K. T. Kalleberg, E. Visser (2012). Software development environments on the web. Presented by the Association for Computing Machinery, 2012.
8. H. B. Salameh, C. Jeffery (2011). Collaborative and social development environments for software development. Published in the International Journal of Computer Applications in Technology, 2011.



9. D. Sun, S. Xia, C. Sun, D. Chen (2010). Operational Transformation for Collaborative Word Processing. Published by the Association for Computing Machinery, 2010.
10. S. Klein, N. Vehring, M. Kramer (2009). Introducing Real-Time Code Compiler Environment. Found in the International Journal of Computer Applications in Technology, 2009.
11. Jingwen Ou, Mahdi Tayarani Najaran, Mushfiqur Rouf (2007). Aurora SDK: A Web-Based Integrated Development Environment. Published in the International Journal of Innovative Technology and Exploring Engineering (IJITEE), 2007.
12. C. W. Fraser (1989). A Language for Writing Code Generators. Presented at the SIGPLAN'89 Conference on Programming Language Design and Implementation, featured in SIGPLAN Notices, volume 24, issue 7, pages 238–245.