iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** V **Month of publication:** May 2025

DOI: https://doi.org/10.22214/ijraset.2025.70799

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 13 Issue V May 2025- Available at www.ijraset.com

DevForge: An Online Coding Platform

Amogh M[1], Abhilash K[2], Thathva B R[3], Vivek R[4], Madhusudan G[5]
CSE, JSS STU, Mysuru, India

***Abstract:** The DevForge project is an innovative online coding platform built to inspire and support programmers at every stage of their learning journey. It brings together multiple essential elements of a modern development environment — a real-time coding playground, a structured problem-solving arena, and a competitive battleground for coding challenges and contests.*

I. INTRODUCTION

In today's digital age, the demand for platforms that encourage hands-on coding, learning through challenges, and healthy competition is greater than ever. With this in mind, the DevForge project was conceived — an innovative online coding platform designed to empower developers, students, and coding enthusiasts of all levels. DevForge offers a dynamic and multifaceted environment where users can explore and sharpen their programming skills. At its core, the platform provides a coding playground for users to write, test, and experiment with code in real-time. It also includes a problem-solving arena, where curated coding challenges help users improve their logical thinking and algorithmic abilities. To add an exciting layer of engagement, DevForge introduces a competitive battleground, allowing users to participate in coding contests, track their progress, and compare performance with peers. What sets DevForge apart is its focus on learning, collaboration, and growth. Whether someone is a beginner trying to grasp the fundamentals or an experienced developer looking to stay sharp, the platform offers tools and features tailored to support continuous development. By fostering a sense of community and friendly competition, DevForge not only enhances individual coding skills but also promotes a collaborative culture within the developer ecosystem. This report presents the design, development, and functional aspects of the DevForge platform, detailing its objectives, features, and the potential it holds in shaping the next generation of developers.

II. LITERATURE REVIEW

With the growing demand for interactive programming environments, the concept of online coding platforms has evolved to support a wide range of educational and collaborative needs. DevForge emerges as a response to this demand—integrating a code editor, problem-solving space, and competitive arena to foster a holistic learning experience.

Sharma et al. [1] highlighted the fundamental utility of online compilers in providing real-time code execution within a browser, paving the way for lightweight, accessible platforms. This groundwork informs DevForge’s core coding environment, which allows users to experiment and execute code seamlessly.

The role of coding platforms in distance education was further analyzed by Zinovyeva et al. [3], who emphasized that such tools enhance engagement and accessibility. Their findings validate DevForge’s design to support learners remotely, particularly through its instant feedback and intuitive interface.

The educational value of competitive programming was underlined by Yuen et al. [2], who demonstrated that it cultivates critical thinking and algorithmic problem-solving. DevForge builds upon this by incorporating structured contests and real-time leaderboards, promoting both skill development and motivation.

Furthering this notion, the Emory University thesis [5] explored how analyzing competitive programming data can lead to more effective training methods. DevForge incorporates similar analytics-driven feedback to help users track and improve their coding performance over time.

Pathak et al. [7] introduced V-Code, an online code editor designed with simplicity and collaboration in mind. Their focus on usability and student engagement aligns with DevForge’s goal of creating a beginner-friendly yet powerful coding tool that facilitates both solo and collaborative work.

Collaborative learning paradigms in software education were examined in a recent SIGCSE study [4], which found that environments encouraging teamwork significantly enhance educational outcomes. DevForge adopts this strategy by enabling collaborative coding challenges and peer feedback systems.

From a data perspective, the IEEE study on collaborative editing [6] revealed that real-time data collection from user interactions could be leveraged to improve platform efficiency and user experience. DevForge employs a similar data-driven approach to personalize learning journeys and optimize platform performance.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue V May 2025- Available at www.ijraset.com

In summary, the existing literature underlines the importance of integrating code execution, collaborative learning, and competition into a single platform. By synthesizing these insights, DevForge aims to be a comprehensive and engaging space for learners, developers, and competitive coders alike.

III. METHODOLOGY

A. Tools and Technology

To build DevForge as a scalable, secure, and user-friendly online coding platform, a range of modern web development tools and technologies were employed. The frontend was developed using React to deliver a responsive user interface, while Node.js powers the backend logic for contest operations and code evaluation. Firebase services were used for real-time data handling, authentication, and persistent storage. For executing user-submitted code securely and efficiently, the Judge0 API was integrated. The following table summarizes each technology and its role in the system.

TABLE I

Tool	Description
React	Frontend framework for building an interactive and responsive UI.
Node.js	Backend runtime to handle business logic, contest management, and API integration.
Firebase Auth	User authentication and authorization service.
Firebase DB	Cloud-hosted NoSQL database to store persistent user data.
Firebase Realtime DB	Enables live updates for coding arenas, contests, and leaderboard changes.
Judge0 API	Open-source API used for language-specific code execution and evaluation.
API Gateway/Load Balancer	Ensures scalability and efficient distribution of frontend requests to backend services.
Analytics Tools	Monitoring and performance tracking across user activities and system logs.

B. Structure

The core structure of DevForge is organized around modular and scalable data representations, particularly optimized for handling programming challenges, user submissions, and real-time interactions. The database schema is primarily managed through Firebase Firestore and Realtime Database, enabling fast read/write operations and live updates during contests. Each coding question is stored with relevant metadata such as difficulty level, category, and video resources to support structured learning. This design also supports user progression tracking and content organization. A representative schema for a coding problem is shown below.

Database Schema:

```
Problems:
        {
          id: String;
          title: String;
          difficulty: String;
```

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

```
    category: String;
    order: Integer;
    videoId: String;
}

Files:
{
  id: string;
  name: string;
  type: 'file';
  content: string;
  input: string;
  output: string;
  language?: string;
  createdAt: string;
  updatedAt: string;
}

Folders:
{
  id: string;
  name: string;
  type: 'folder';
  children: FileNode[];
}
```

C. System Implementation

The voting system has a three-layer architecture, which renders it modular, scalable, and maintainable.

1) *FRONTEND:* The user interface of the platform is developed using React, a popular JavaScript library for building dynamic web applications. The frontend includes multiple interactive modules designed to enhance the learning and competitive coding experience:

Playground: A free-form coding environment that allows users to write, compile, and execute code in various programming languages. This functionality is powered by the Judge0 API, enabling real-time code evaluation directly in the browser.

Coding Arena: This module presents users with data structures and algorithms (DSA) problems. It supports user code submissions and validates them against predefined input and output test cases to assess correctness.

Battleground: A competitive module that hosts real-time contests where participants can solve challenges within a time limit. The feature includes real-time tracking of performance through a dynamic leaderboard.

To manage authentication, Firebase Authentication is integrated into the frontend. This enables secure sign-up, login, and session management for all users.

2) *BACKEND:* The backend of the system is developed using Node.js and leverages Firebase services for data management and real-time communication. It performs the following key functions:

Playground Compiler Service: This service receives code submissions from the frontend, forwards them to the Judge0 API for compilation and execution, and sends the results back to the user interface.

Challenge Management (Coding Arena & Battleground): This component handles the logic behind timed coding problems and contests. It records user submissions, evaluates their correctness, and updates their scores accordingly. It also enforces timing and problem constraints for a fair competition environment.

Database Integration: All user activity, problem data, and results are stored in Firebase's database services. For real-time features such as leaderboard updates and submission tracking, the system uses Firebase Realtime Database to instantly reflect changes across all connected clients.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

API Gateway: An API Gateway is used to route requests to appropriate backend services and to balance the load during high traffic, especially during live contests.

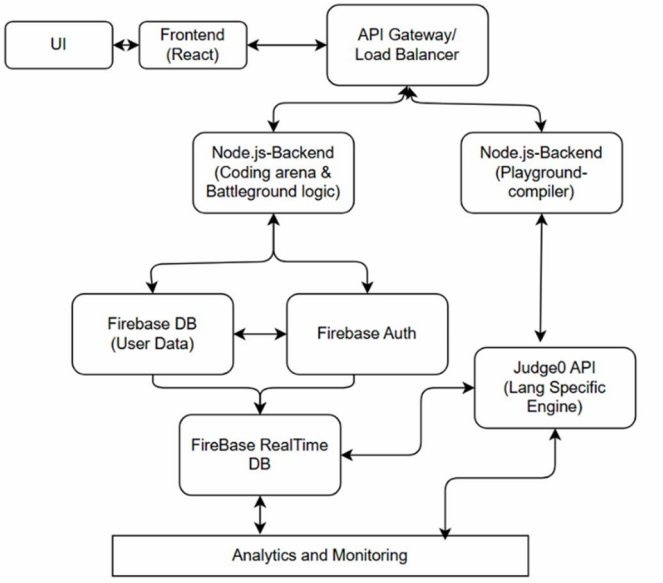


Fig.1 Data Flow Diagram

IV. RESULTS AND DISCUSSION

A. *System Performance and Security Evaluation*

The system ensures minimal latency during code compilation using asynchronous request handling with Judge0 API.
Real-time database updates and leaderboard refreshes occur within milliseconds, enabling a seamless user experience in contests.
Firebase Auth enforces secure access, preventing unauthorized access to problem sets or results.

B. *Key Performance Metrics*

Metric	Value (Avg.)
Compilation Latency	~1.2 seconds
Real-time Sync Delay	< 300 milliseconds
Authentication Time	~800 milliseconds
Uptime	> 99.9% (Firebase SLA)

C. *Security Architecture*

Authentication: Firebase Auth with email/password and OAuth (Google) sign-in methods.
Data Security: Firebase Firestore and Realtime DB enforce role-based access using Firestore security rules.
API Protection: Node.js backend enforces API keys for internal routes and secures Judge0 endpoints from abuse.
Traffic Management: API Gateway restricts unusual activity and rate-limits traffic to mitigate DDoS threats.

D. *Additional System Features*

Video Hints: Each question includes a video explanation for guided learning.
Question Ordering: Problems are displayed based on difficulty and progression.
Analytics Dashboard: Admins can track user activity, submissions, and leaderboard dynamics.
Extensibility: The platform can be expanded to include user-generated content and community challenges.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

V. CONCLUSION

The DevForge project signifies a transformative leap in the realm of online coding platforms by offering a rich, interactive environment where learning, practice, and competition coexist seamlessly. It bridges the gap between theory and practical application through its real-time coding playground, curated challenge arena, and competitive battleground, ensuring users are not only learning syntax but also developing problem-solving abilities, algorithmic thinking, and coding efficiency. Designed with inclusivity at its core, DevForge supports learners of all levels—from absolute beginners gaining confidence to advanced developers refining their expertise—while promoting a growth-oriented, user-friendly experience. Its community-driven approach encourages collaboration, peer-to-peer learning, and healthy competition, creating a vibrant ecosystem where users can engage meaningfully and stay motivated. By enabling user-generated content, tracking performance metrics, and supporting multiple languages and development tools, the platform goes beyond traditional teaching models and fosters innovation, adaptability, and self-directed learning. DevForge is not just a coding platform; it is a comprehensive educational space that cultivates technical skillsets, boosts digital literacy, and prepares users for real-world software development challenges. In an era where coding proficiency is becoming as fundamental as literacy, DevForge stands poised to empower the next generation of problem-solvers, engineers, and innovators by nurturing talent in a supportive, scalable, and forward-looking digital environment.

REFERENCES

[1] Sharma, A., Dixit, A., & Upadhyay, B. (2017). Online compiler [Journal-article]. International Research Journal of Engineering and Technology (IRJET), 239–240. https://www.irjet.net/archives/V4/i5/IRJET-V4I542.pdfS. Shukla, A. N. Thasmiya, D. O. Shashank, and H. Mamatha, ”Online voting application using Ethereum blockchain,” 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 873–880, 2018.

[2] Yuen, Kevin Kam Fung & Liu, Dennis & Leong, Hong. (2023). Competitive programming in computational thinking and problem solving education. Computer Applications in Engineering Education. 10.1002/cae.22610. D. Raikar and A. Vatsa, ”BCT-voting: A blockchain technology-based voting system,” The 27th International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA’21), pp. 26–29, 2021.

[3] Zinovyeva, Irina & Artemchuk, Volodymyr & Iatsyshyn, Anna & Popov, O & Valeriia, Kovach & Andrii, Iatsyshyn & Radchenko, Oleksandr. (2021). The use of online coding platforms as additional distance tools in programming education. Journal of Physics: Conference Series. 1840. 012029. 10.1088/1742-6596/1840/1/012029. Ahmed Ben Ayed, ”A conceptual secure blockchain-based electronic voting system”, International Journal of Network Security & Its Applications, vol. 9, no. 3, pp. 01-09, 2017.

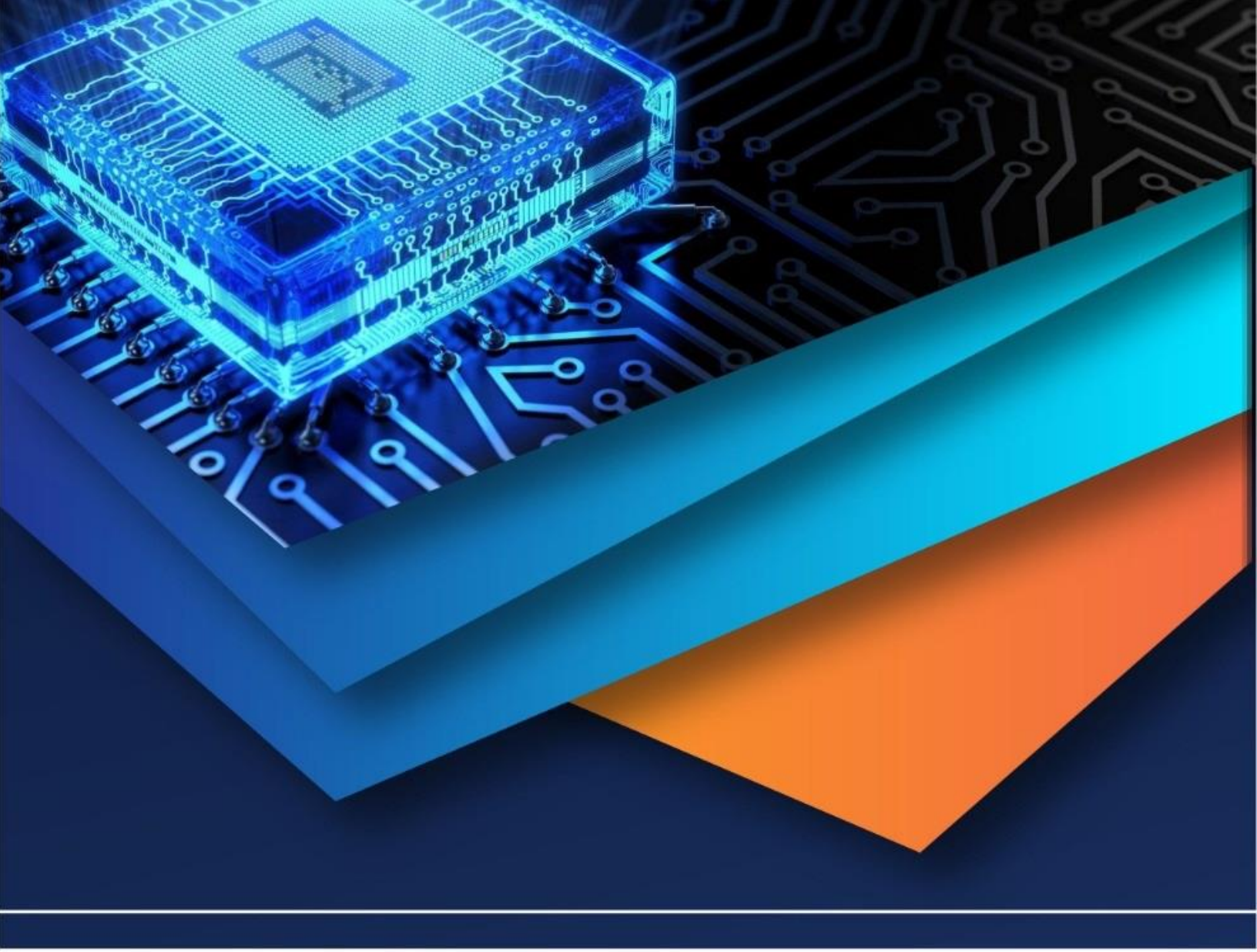
[4] Application of Collaborative Learning Paradigms within Software Engineering Education: A Systematic Mapping Study (SIGCSE TS 2024 - Papers) - SIGCSE TS 2024. (n.d.). https://sigcse2024.sigcse.org/details/sigcse-ts-2024-Papers-1/1/Application-of-Collaborative-Learning-Paradigms-within-Software-Engineering-EducationM. Hellman, Yavuz Emre, Ali Kaan Koc¸, Umut Can C¸abuk and Gokhan¨ Dalkihc¸, ”Towards secure e-voting using ethereum blockchain”, 2018 6th International Symposium on Digital Forensic and Security (ISDFS), pp. 1-7, 2018.

[5] ETD | Analyzing Competitive Programming Competitions to Develop Effective Training Methods for Improving Problem-Solving Skills in Students | ID: s4655h97f | Emory Theses and Dissertations. (n.d.). https://etd.library.emory.edu/concern/etds/s4655h97fCaiazzo, Francesca, and Ming Chow. ”A block-chain implemented voting system.” Computer System Security 1.1 (2016): 1-13.

[6] Gathering useful programming data; analysis and insights from real-time collaborative editing. (2015, May 1). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/7160270Amores-Sesar, Ignacio, Christian Cachin, and Enrico Tedeschi. ”When is spring coming? A security analysis of avalanche consensus.” arXiv preprint arXiv:2210.03423 (2022).

[7] Pathak, P. R., Magade, T. V., Vichare, A. A., Repale, S. I., Department of Information Technology, Pillai HOC College of Engineering and Technology, & Department of Information Technology, Pillai HOC College of Engineering and Technology, Maharashtra, India. (n.d.). V-Code: Online Code Editor. In International Journal for Research Trends and Innovation (Vol. 8, Issue 4, pp. 1005–1006) [Journal-article]. https://www.ijrti.org/papers/IJRTI2304164.pdfKhan, Maliha, et al. ”Face detection and recognition using OpenCV.” 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS). IEEE, 2019.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



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)