STM JOURNALS

Recent Trends in
Programming Languages

ISSN: 2455-1821
Volume 11, Issue 2, 2024
May–August
DOI (Journal): 10.37591/RTPL

https://journals.stmjournals.com/rtpl

Review RTPL

Agile Modeling and Building an Inclusive Online Coding Platform for Programming

Kiran Waghmare[1,*], Viraj Tandel[2], Ankita Pareek[2]

Abstract

The integration of agile modeling principles within professional programming emphasizes the development of online coding platforms tailored to meet the diverse needs of computer science and engineering students. This study introduces an agile-based, API-integrated J2EE online coding platform, designed to create a sophisticated and user-friendly environment for seamless programming practice, particularly focusing on Java programming. The platform offers real-time tracking of programming progress and enhances the understanding of complex coding concepts, fostering a collaborative and interactive learning experience between teachers and students. A key feature of the platform is its ability to facilitate the identification and comprehension of intricate coding concepts by presenting unexpected input values during practice sessions. This not only challenges students to think critically and problem-solve but also enhances their ability to write high-quality programs. By emphasizing agility and inclusivity in designing educational technology solutions, the platform aims to enhance the quality of coding-related coursework. It equips students with the essential skills and knowledge to succeed in their programming efforts. Ultimately, this agile-based online coding platform aims to provide an innovative and effective solution for modernizing computer science education, making it more accessible, engaging, and effective for all learners. In conclusion, this platform represents a significant advancement in educational technology, offering a robust and scalable approach to teaching programming. It prepares students for real-world challenges, ensuring they are well-equipped with the skills required for a successful career in technology.

Keywords: Agile modeling, online coding platform, J2EE, Springboot technology, cloud

INTRODUCTION

In today's digital age, computer science education plays a pivotal role in preparing individuals for the rapidly evolving technological landscape. As emphasized by Wing, computational thinking has emerged as a fundamental skill set, enabling individuals to enhance their analytical capacity and develop abstract, algorithmic, logical, and scalable thinking abilities [1]. The significance of computational thinking transcends the boundaries of computer science, extending its influence on various domains of knowledge, as highlighted by Brennan and Resnick [2]. This transdisciplinary nature of computational thinking underscores its importance in fostering reflective and critical thinking among young learners, thus equipping them with essential skills for navigating the complexities of the modern world.

However, the teaching of computer science, particularly programming-related courses, presents unique challenges. Large class sizes limit the amount of personal attention students receive from teachers, hindering their ability to grasp complex

***Author for Correspondence**
Kiran Waghmare
E-mail: kdkiran2003@gmail.com

[1]Programme Manager, Department of Education and Training, Centre for Development of Advanced Computing (CDAC), Mumbai, Maharashtra, India
[2]Student, Department of Education and Training, Centre for Development of Advanced Computing (CDAC), Mumbai, Maharashtra, India

Received Date: June 30, 2024
Accepted Date: July 31, 2024
Published Date: August 07, 2024

Citation: Kiran Waghmare, Viraj Tandel, Ankita Pareek. Agile Modeling and Building an Inclusive Online Coding Platform for Programming. Recent Trends in Programming Languages. 2024; 11(2): 33–42p.

concepts and apply theoretical knowledge to practical problems [3]. Moreover, the rapid progression through Bloom's taxonomy required in programming courses necessitates hands-on practice and real-world application to ensure limitless learning and skill development.

Online learning increases the challenges, disrupting traditional classroom settings and necessitating innovative approaches to practical learning. In response, there is a growing recognition of the need for online platforms to facilitate coding practical and provide students with accessible and effective learning experiences [4]. This work proposes the development of an agile-based online coding platform to enhance practical learning in computer science education.

Drawing upon the critical role of teaching methods and the growing importance of programming skills, this work outlines the rationale behind the proposed agile-based online coding platform. By providing seamless access to programming tools and resources, the platform aims to overcome the limitations of traditional classroom settings and empower students to engage in hands-on coding practice from any location [5]. Furthermore, the platform's integration with automated assessment tools will enable instructors to track student progress and provide personalized ranking, thus enhancing the effectiveness and accessibility of practical learning in computer science education [6].

The following sections of this paper are organized as follows: Next Section provides an overview of related work concerning the teaching of coding-related courses and examines existing coding platforms with a focus on agile modeling principles. The Section after that delves into the implementation details of our proposed platform, highlighting the integration of agile modeling methodologies. In the Section afterwards, we present the methodology defined for the use cases and implementation details, about the implemented platform. Also, the following Section discusses the results and test cases executed for the online platform. Finally, concludes the paper by summarizing key findings and discussing implications for future research and practice.

RELATED WORK

The demand for modern and effective tools to enhance the quality of education and develop technical skills, especially in the field of computer science, is increasingly recognized both globally and in specific contexts like Ukraine [7]. They have highlighted the importance of diversifying teaching methods, particularly in programming education, to meet the needs of diverse learners and improve learning outcomes as shown in Figure 1 [8].

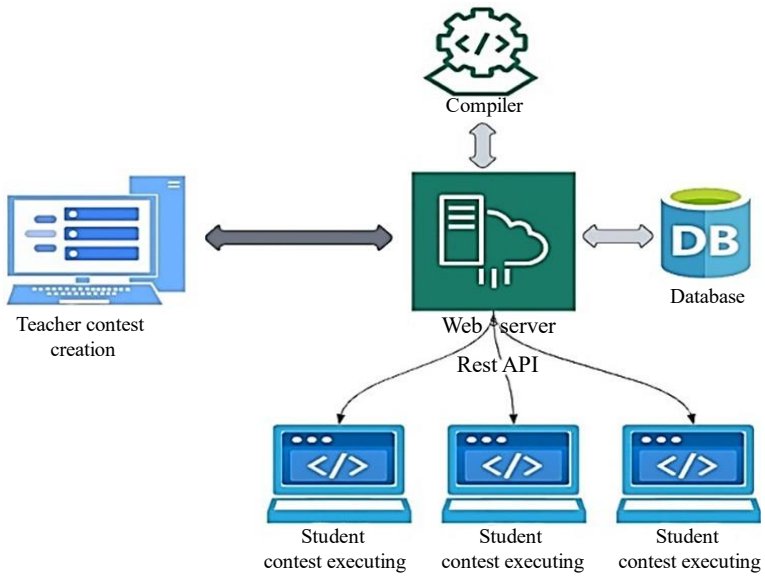


Figure 1. Architecture diagram for agile-based online coding platform.

Several studies have explored innovative approaches to teaching programming, focusing on strengthening students' programming skills and fostering their innovation and problem-solving abilities. For instance, Sun *et al.* discuss the importance of student-centered teaching methods and the integration of theory and practice to enhance learning efficiency and stimulate student's interest in programming [9]. Their reform activities emphasize task-driven and situational teaching, which has shown promising results in developing students' independent programming design skills.

Similarly, Zhao *et al*. [10] and Shen *et al.* [11] have investigated web-based cooperative learning and online collaboration examinations as effective teaching strategies for coding-related courses. These approaches aim to enhance student engagement and improve teaching effectiveness through interactive and collaborative learning environments.

Agile methodology emerges as the most promising approach for the development of online coding platforms due to its iterative and collaborative nature, aligning seamlessly with the dynamic demands of programming education. Unlike traditional waterfall methods, agile allows for continuous feedback, enabling rapid adaptation to evolving student needs and technological advancements. By breaking down development into manageable sprints, agile ensures that features are delivered incrementally, fostering greater flexibility and responsiveness to user requirements [12]. Moreover, Agile promotes close collaboration between developers, educators, and students, facilitating the integration of personalized features and interactive learning experiences. This iterative and collaborative approach not only accelerates the platform's development but also enhances its functionality and usability over time. As online coding platforms strive to provide a rich and engaging learning environment, agile methodology emerges as the optimal strategy to meet the evolving challenges and demands of programming education effectively in online manure.

AGILE MODELLING FOR ONLINE CODING PLATFORM

Agile Modelling

In the realm of online education, agile modeling principles offer a dynamic framework for the design and development of online coding platforms that can adapt to the evolving needs of students and teachers. Agile methodologies prioritize iterative development, collaboration, and responsiveness to change, making them particularly well-suited for the rapid pace and diverse requirements of programming education. The features of an agile-based online coding platform are as follows:

- *Iterative development:* Features are incrementally built, tested, and refined based on continuous user feedback, allowing for gradual enhancement.
- *User-centered design:* Involving students and teachers in development ensures the platform meets their specific needs, aligning with learning objectives.
- *Collaborative development:* Collaboration among developers, educators, and designers brings diverse perspectives, enhancing learning experiences.
- *Flexibility and adaptability:* Platforms can quickly respond to changes in programming education, incorporating new features and technologies.
- *Continuous improvement:* Emphasizes delivering value through regular iterations and enhancements, evolving to meet evolving needs.

Incorporating Agile modeling principles into the design and development of online coding platforms holds significant promise for enhancing the effectiveness and usability of these platforms in supporting programming education. By embracing iterative development, user-centered design, collaborative development, flexibility, and continuous improvement, online coding platforms can better serve the needs of learners and educators in the dynamic and rapidly evolving field of programming education.

Architecture for Online Coding Platform

The Agile model architecture for the online coding platform prioritizes adaptability, teamwork, and swift responses to evolving requirements. It commences by gathering user stories and requirements

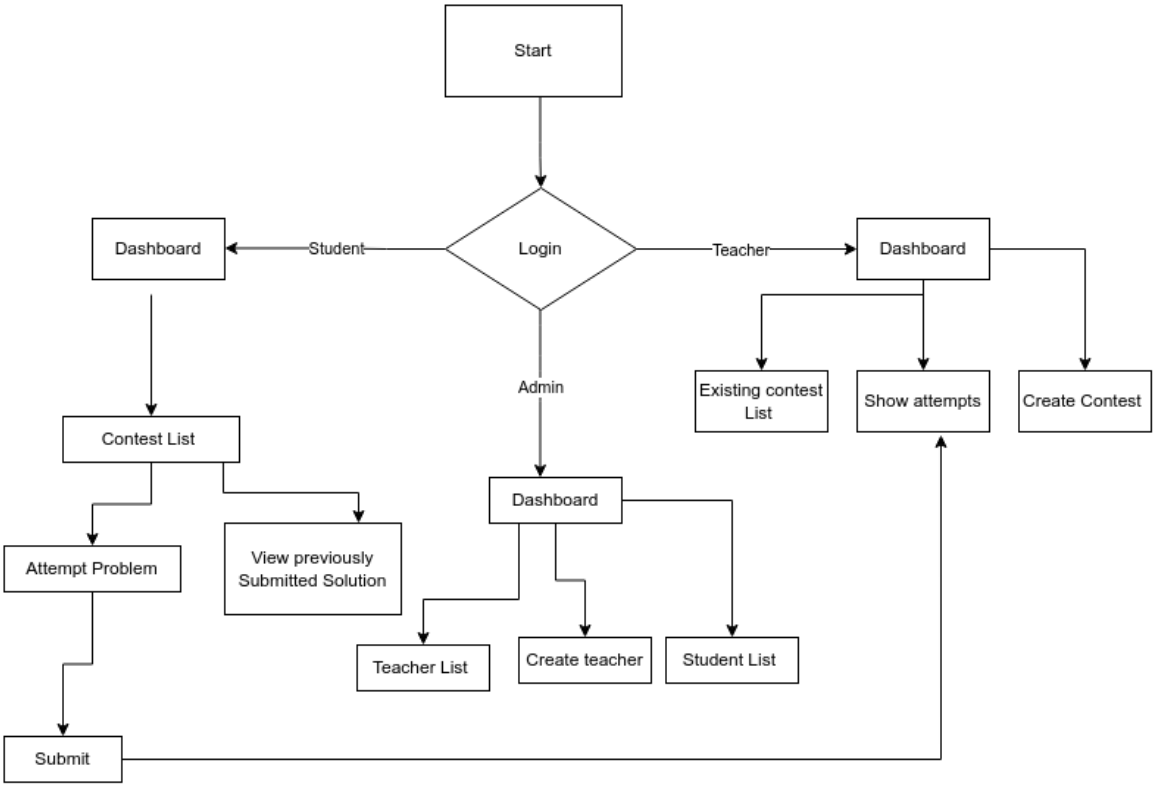


Figure 2. Flow diagram for Agile-based online coding platform.

from stakeholders such as teachers, students, and administrators. Agile teams, comprising diverse skill sets including developers, designers, and testers, collaborate closely to implement these requirements in iterative sprints lasting 1–4 weeks. During each sprint, the team focuses on achieving specific goals defined at the outset, with daily stand-up meetings ensuring alignment and coordination among members.

Continuous integration and testing practices are integral to the Agile model, facilitating the seamless integration of code changes and ensuring the reliability of the platform. Following each sprint, stakeholders participate in review meetings to assess completed features and provide feedback. These insights are then used to refine and adapt development processes in sprint retrospectives, fostering a culture of continuous improvement. Overall, the Agile approach emphasizes delivering functional software increments at the end of each sprint, enabling rapid iteration, stakeholder involvement, and sustained product enhancement.

The proposed system works on the agile model and the flow diagram illustrates the interactions and pathways for students, teachers, and developers within the system, as given in Figure 2. It delineates the sequence of actions and decision points that each role encounters while navigating the platform. Through intuitive design and clear visualization, the diagram offers a comprehensive overview of how users engage with the system, facilitating understanding and guiding stakeholders through their respective roles and tasks.

Use Cases for Agile Methodology

User stories help to articulate the needs and requirements of different users concisely and understandably. Here are some user stories for an online coding platform using Agile modeling:

- As a student, I want to have access to a variety of coding exercises and challenges that are incrementally challenging, so I can continuously improve my programming skills at my own pace.
- As a teacher, I want to be able to gather feedback from students on the usability and effectiveness of the platform, so I can make informed decisions about improvements and updates.

- As a developer, I want to collaborate with educators and instructional designers to ensure that the platform is user-friendly, aligned with learning objectives, and incorporates best practices in coding education.
- As a student, I want the platform to be flexible and adaptable to changes in programming education, so I can stay up-to-date with emerging trends and technologies in the field.
- As a teacher, I want to be able to track students' progress and performance on coding exercises, so I can provide timely feedback and support to help them succeed.
- As a student, I want the platform to foster a sense of community and collaboration among peers, so I can learn from others, share knowledge, and work together on coding projects.
- As a developer, I want to continuously improve the platform based on user feedback and performance metrics, so it remains relevant, effective, and valuable to both students and teachers.

These user stories represent the needs and expectations of different stakeholders involved in the development and use of an online coding platform. They serve as a guide for prioritizing features and functionalities during the Agile development process.

Proposed Technology

The technologies involved in the operating environment of the project include as following:

- *Hardware:* A machine with at least 8 GB of RAM: Ensures sufficient memory for running multiple applications concurrently without performance degradation. Fast processor (e.g., Intel Core i5 or higher): Provides the computational power needed for efficient execution of tasks, such as compiling code, running servers, and processing user requests.
- *Software:* Code Editor: Utilized for writing, editing, and managing source code files. Common options include Visual Studio Code, Eclipse, IntelliJ IDEA, or Sublime Text.
- *ReactJS:* Framework for building user interfaces (UIs) in JavaScript. Used for front-end development to create interactive and responsive web applications with reusable components.
- *Spring boot:* Framework for building Java-based backend applications. Provides a streamlined way to create standalone, production-grade Spring-based applications with minimal setup and configuration.
- *MySQL:* Relational database management system (RDBMS) used for storing and managing application data. MySQL is a popular choice for its reliability, performance, and ease of use, making it suitable for a wide range of applications, including web development projects.

By leveraging these technologies, developers can create a robust and scalable web application with a responsive user interface, efficient backend logic, and reliable data management capabilities. This technology stack offers a powerful combination of tools and frameworks to meet the project's requirements for performance, scalability, and maintainability.

METHODOLOGY

The methodology used to develop the software application consists of a server-client API integration method. It details the sprint goal, implementation, and the expected output are as follows:

User Registration and Login

- *Sprint goal:* Implement user registration and login functionality for both teachers and students.
- Develop user registration forms, implement an authentication system, and create login pages as shown in Figure 3.
- *Expected output:* Users can create accounts and log in to access platform features.

Contest Creation and Management

- *Sprint goal:* Enables teachers to create and manage contests for students.
- Implement class creation forms, allow teachers to assign passwords for contest enrollment, and develop features for monitoring student progress as shown in Figure 4.

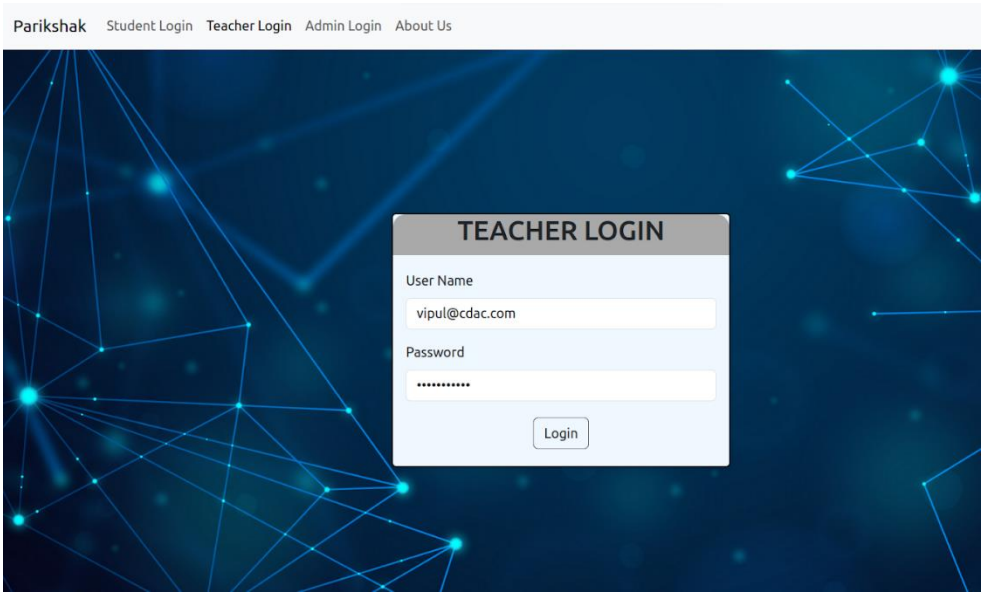


Figure 3. Teacher login.

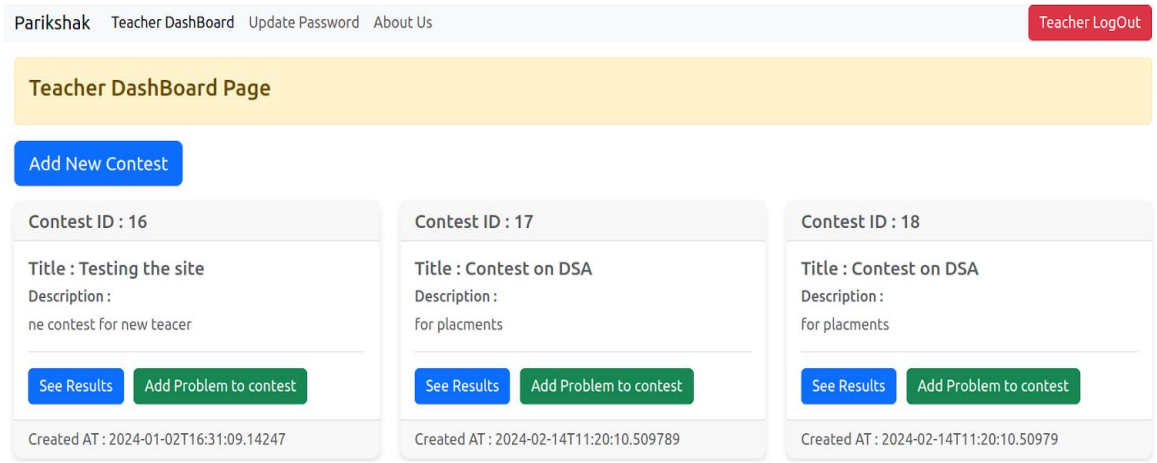


Figure 4. Contest creation and management.

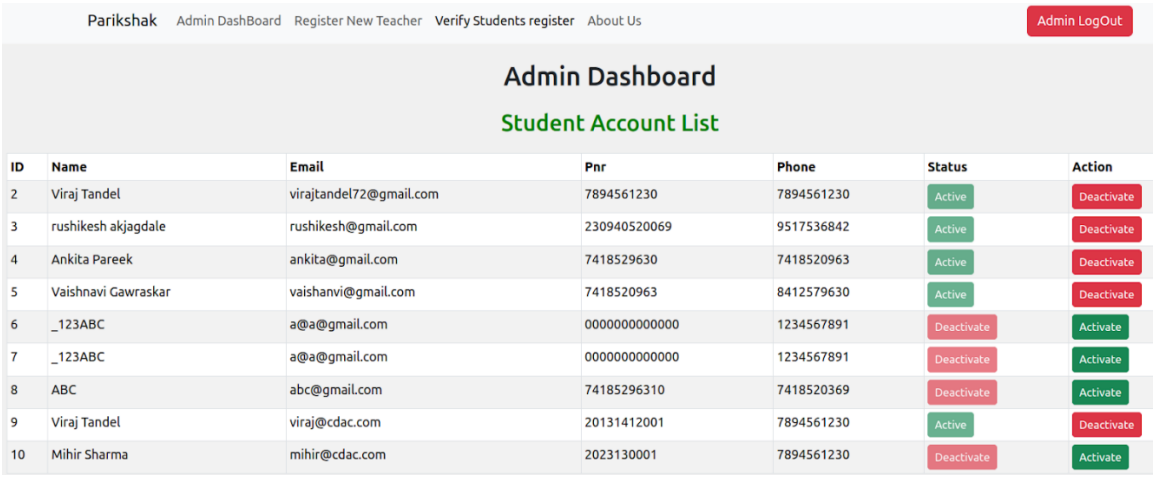
- *Expected output:* Teachers can create contests, distribute enrollment passwords, and track student activities within each class.

Code Compilation and API Management

- *Sprint goal:* Implement a code editor with a runtime analyzer and integrate API for code compilation and evaluation.
- Develop a code editor with multiple language support, integrate runtime analyzer, and implement API for code compilation and evaluation as shown in Figure 5.
- *Expected output:* Students can write and execute code within the platform, receive immediate feedback on their submissions, and have the results stored in the database.

Database Management

- *Sprint goal:* Set up a relational database to store user, class, and submission information.
- Design database schema, implement data storage and retrieval functionality, and ensure data integrity and security.
- *Expected output:* All relevant information, including user accounts, class details, and programming submissions, is securely stored and managed in the database.



ID	Name	Email	Pnr	Phone	Status	Action
2	Viraj Tandel	virajtandel72@gmail.com	7894561230	7894561230	Active	Deactivate
3	rushikesh akjagdale	rushikesh@gmail.com	230940520069	9517536842	Active	Deactivate
4	Ankita Pareek	ankita@gmail.com	7418529630	7418520963	Active	Deactivate
5	Vaishnavi Gawraskar	vaishanvi@gmail.com	7418520963	8412579630	Active	Deactivate
6	_123ABC	a@a@gmail.com	0000000000000	1234567891	Deactivate	Activate
7	_123ABC	a@a@gmail.com	0000000000000	1234567891	Deactivate	Activate
8	ABC	abc@gmail.com	74185296310	7418520369	Deactivate	Activate
9	Viraj Tandel	viraj@cdac.com	20131412001	7894561230	Active	Deactivate
10	Mihir Sharma	mihir@cdac.com	2023130001	7894561230	Deactivate	Activate

Figure 5. Dashboard for contest.

Work Tracking

- *Sprint goal:* Develop a system for tracking students' submissions and progress.
- Implement features for students to view their submissions, create dashboards for monitoring progress, and integrate a notification system for upcoming deadlines as shown in the flow diagram of Figure 4.
- *Expected output:* Students can easily track their submissions and progress within each class, ensuring they stay on top of their assignments.

EXPERIMENTATION AND RESULTS

The system was tested to evaluate its functionality in creating practical problems for students. During the testing phase, both teachers and students were able to log in successfully using their credentials. Once logged in, teachers were able to access the functionalities of the system related to creating practical problems. Creating a contest involved three mandatory steps, each of which was tested thoroughly:

1. *Uploading problem statement:* Teachers were provided with a markup-supported editor where they could type the problem statement. During testing, it was ensured that teachers could enter the problem statement accurately and include necessary details such as sample inputs, outputs, constraints, and explanations for samples if required as shown in Figure 6.

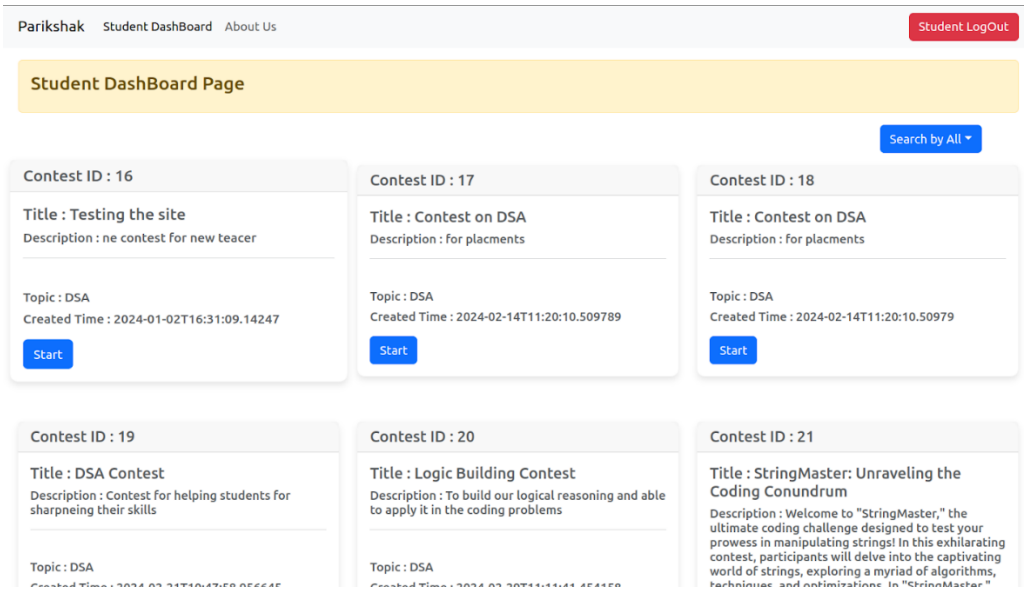


Figure 6. Dashboard for online coding contest for student.

2. *Uploading test data:* In this step, teachers upload test cases (inputs and corresponding expected outputs) to judge students' codes. Each test case was uploaded separately, and teachers had the option to allow different weights to each test case. Default weightage, set to equal distribution of marks for each test case, was also tested for accuracy with different difficulty levels i.e. easy, medium, and hard problem statements as shown in Figure 7.
3. *Testing:* After uploading the test data, teachers had to make a successful submission to validate the uploaded test data. During testing, it was ensured that the submission process functioned smoothly and that teachers could verify the correctness and effectiveness of the test data by submitting various expected wrong or inefficient approaches as shown in Figure 8.

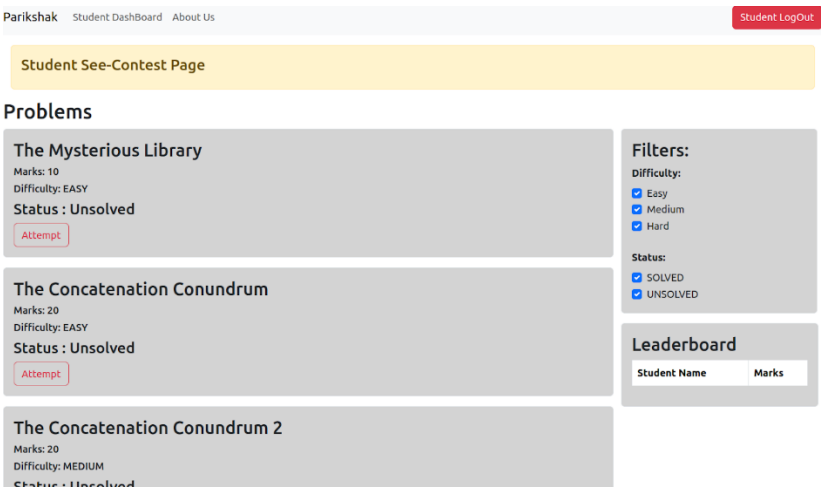


Figure 7. Dashboard for uploaded coding questions online coding contest for a student.

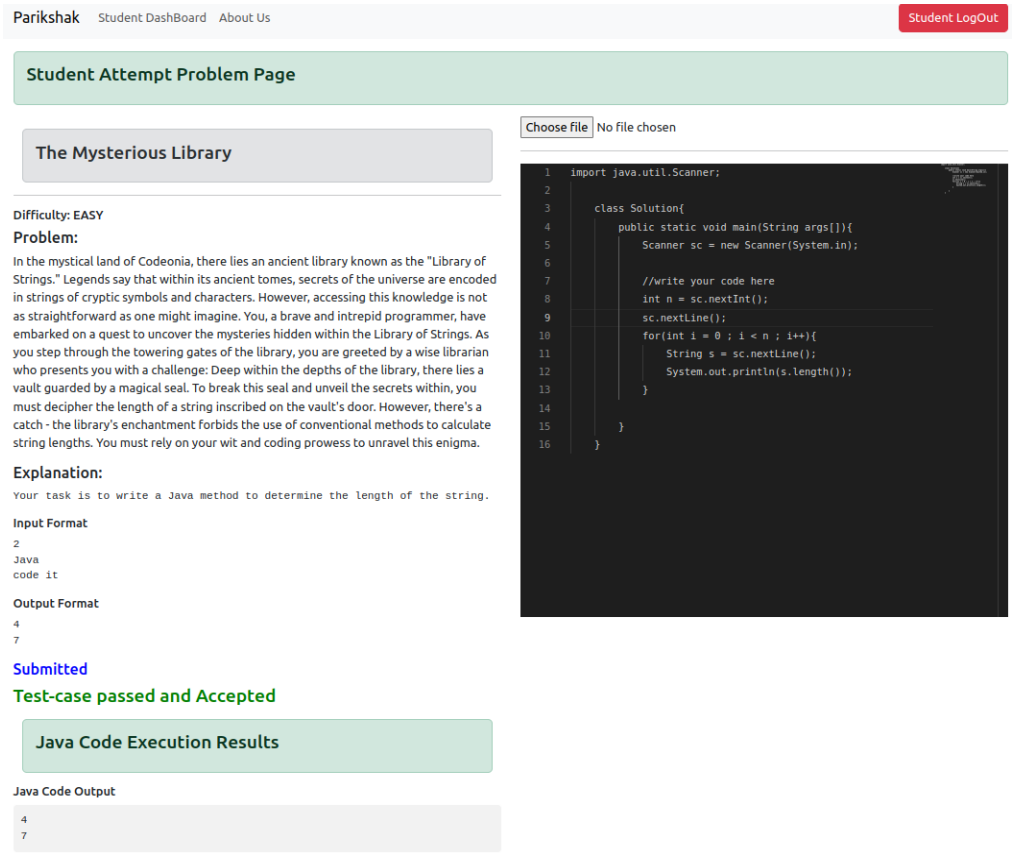
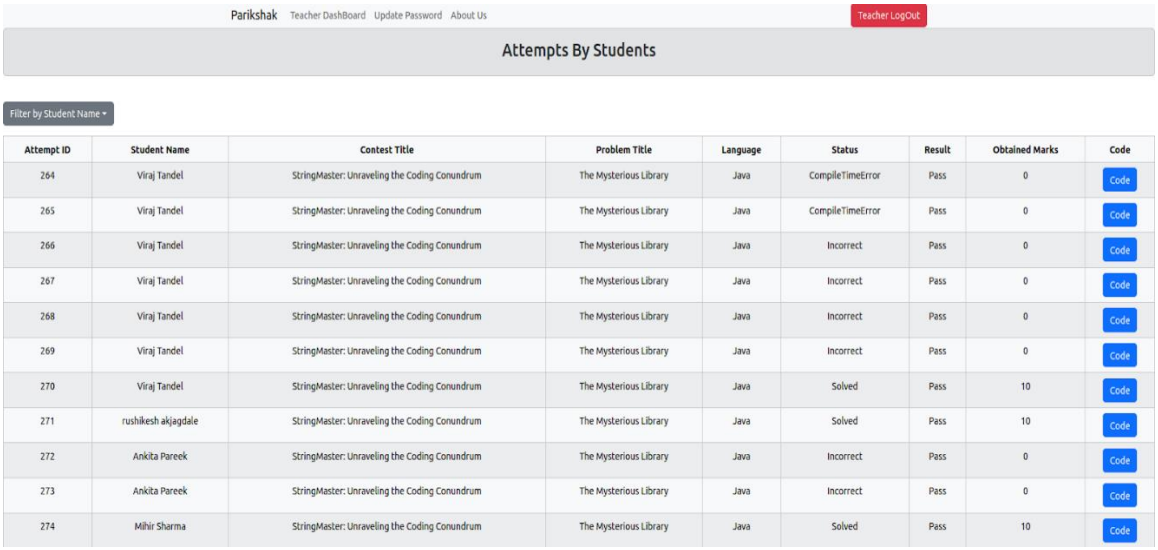


Figure 8. Agile-based online coding contest with executable test cases.



Attempt ID	Student Name	Contest Title	Problem Title	Language	Status	Result	Obtained Marks	Code
264	Viraj Tandel	StringMaster: Unraveling the Coding Conundrum	The Mysterious Library	Java	CompileTimeError	Pass	0	Code
265	Viraj Tandel	StringMaster: Unraveling the Coding Conundrum	The Mysterious Library	Java	CompileTimeError	Pass	0	Code
266	Viraj Tandel	StringMaster: Unraveling the Coding Conundrum	The Mysterious Library	Java	Incorrect	Pass	0	Code
267	Viraj Tandel	StringMaster: Unraveling the Coding Conundrum	The Mysterious Library	Java	Incorrect	Pass	0	Code
268	Viraj Tandel	StringMaster: Unraveling the Coding Conundrum	The Mysterious Library	Java	Incorrect	Pass	0	Code
269	Viraj Tandel	StringMaster: Unraveling the Coding Conundrum	The Mysterious Library	Java	Incorrect	Pass	0	Code
270	Viraj Tandel	StringMaster: Unraveling the Coding Conundrum	The Mysterious Library	Java	Solved	Pass	10	Code
271	rushikesh akjagdale	StringMaster: Unraveling the Coding Conundrum	The Mysterious Library	Java	Solved	Pass	10	Code
272	Ankita Pareek	StringMaster: Unraveling the Coding Conundrum	The Mysterious Library	Java	Incorrect	Pass	0	Code
273	Ankita Pareek	StringMaster: Unraveling the Coding Conundrum	The Mysterious Library	Java	Incorrect	Pass	0	Code
274	Mihir Sharma	StringMaster: Unraveling the Coding Conundrum	The Mysterious Library	Java	Solved	Pass	10	Code

Figure 9. Results of Agile-based online coding contest.

Overall, the experimental results demonstrated that the system effectively facilitated the creation of practical programming problems by providing a user-friendly interface for teachers to input problem statements, upload test data, and validate the uploaded data. The testing phase validated the functionality and accuracy of each step in the problem-creation process, ensuring that teachers could create high-quality and comprehensive practical problems for student assessment as shown in Figure 9.

The data gathered from student feedback indicates a high level of satisfaction and enjoyment with the newly implemented online coding platform. Specifically, more than 80% of students expressed the likelihood of the platform. This positive reception underscores the platform's effectiveness in facilitating learning and engagement among students.

Overall, the results suggest that the online coding platform offers significant benefits to both teachers and students. It provides a cost-effective, convenient, and flexible approach to learning coding-related courses in universities and colleges. By leveraging modern technology and interactive features, the platform enhances the educational experience and contributes to the effective teaching and learning of coding skills in academic settings.

CONCLUSION

In conclusion, the agile-based online coding platform has demonstrated its effectiveness in enhancing the learning experience for both students and teachers. The high level of satisfaction and enjoyment expressed by over 80% of students across the platform's success in facilitating engagement and learning. By offering a user-friendly interface and features such as progress tracking and plagiarism detection, the platform addresses common challenges faced in coding education.

Moreover, the platform's cost-effective, convenient, and flexible approach makes it a valuable tool for universities and colleges seeking to modernize their teaching methods. It leverages modern technology and interactive features to create an optimal learning environment for coding-related courses.

Overall, the agile-based online coding platform represents a significant step towards the digital revolution in the education sector. It provides an innovative solution to the challenges of teaching and learning programming, offering benefits such as improved student engagement, better tracking of progress, and motivation for students to explore new programming techniques. With its consistent and error-free performance, the platform has the potential to transform coding education and empower both students and teachers in their programming journey.

Declaration of Interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript.

Acknowledgment

We extend our sincere appreciation to CDAC Mumbai for providing the necessary resources and support for this research endeavor.

REFERENCES

1. Wing JM. Computational thinking. Commun ACM. 2006 Mar 1; 49(3): 33–35.
2. Brennan K, Resnick M. New frameworks for studying and assessing the development of computational thinking. In Proceedings of the 2012 annual meeting of the American educational research association, Vancouver, Canada. 2012 Apr 13; 1: 25.
3. Yeni S, Grgurina N, Saeli M, Hermans F, Tolboom J, Barendsen E. Interdisciplinary integration of computational thinking in K-12 education: A systematic review. Inform Educ. 2024; 23(1): 223–278.
4. Boske C, Elue C, Osanloo AF, Newcomb WS. Promoting inclusive holistic graduate admissions in educational leadership preparation programs. Front Educ. 2018 Apr 5; 3: 17. Frontiers Media SA.
5. Saleh MA, Khan MI, Banerjee S, Safi F. A tale of online learning during COVID-19: A reflection from the South Asian Association for Regional Cooperation (SAARC) countries. Heliyon. 2023 Jun 1; 9(6): e16347.
6. Newkirk J. Introduction to agile processes and extreme programming. In Proceedings of the 24th international conference on Software engineering. 2002 May 19; 695–696.
7. Figueira Á, Leal JP, Paiva JC. Automated Assessment in Computer Science Education: A State-of-the-Art Review. ACM Trans Comput Educ. 2022; 22(3): 34(40p). Available from https://repositorio.inesctec.pt/items/0624643c-52c0-40dd-b4b2-e7a5b99fbb28
8. Gonge SS, Ghatol AA. Education technology used for improving learning skills of computer science and engineering students. In 2013 IEEE International Conference in MOOC, Innovation and Technology in Education (MITE). 2013 Dec 20; 100–103.
9. Sun H, Liu T. Student-Centered Online Teaching Practices in Theoretical Mechanics. High Educ Stud. 2021; 11(2): 233–239.
10. Zhao J, Guo H, Jiang N. Web-Based Cooperative Learning and Application Solution Research. In 2009 IEEE 1st International Workshop on Education Technology and Computer Science. 2009 Mar 7; 1: 377–381.
11. Shen J, Hiltz SR, Bieber M. Learning strategies in online collaborative examinations. IEEE Trans Prof Commun. 2008 Mar 12; 51(1): 63–78.
12. Kerimbayev N, Umirzakova Z, Shadiev R, Jotsov V. A student-centered approach using modern technologies in distance learning: a systematic review of the literature. Smart Learn Environ. 2023 Nov 15; 10(1): 61.

