THE PATENTS

ACT, 1970

COMPLETE

SPECIFICATION

SECTION 10

TITLE

ONLINE CUSTOM CODE EVALUATION PLATFORM

APPLICANT

Mr. S. Dhinakaran

INVENTORS

Mr. M. Sanjay, Mr. R. Rishi, Mr. I. Mohamed Rafil, Mr. L. V. Vishal

Department of Artificial Intelligence and Data Science,

Sri Shakthi Institute of Engineering and Technology, Coimbatore

ABSTRACT:

In the modern era of digitized education and recruitment, evaluating programming skills efficiently, fairly, and at scale has become a major challenge. The traditional approach of manual code assessment is time-consuming, error-prone, and lacks the scalability required to cater to a large number of users. To address these limitations, this project introduces an **Online Custom Code Evaluation Platform** — a secure, modular, and extensible system designed to automate the assessment of programming assignments and challenges in academic and industrial settings.

The platform enables users, such as students and job candidates, to write and submit code in a built-in editor and receive immediate, detailed feedback. It supports multiple programming languages, including C, C++, Java, and Python. Submitted code is executed in a secure, sandboxed environment using Docker containers to ensure complete isolation from the host system and to prevent any form of exploitation or abuse.

The backend of the system is powered by RESTful APIs built using Flask (Python) or Spring Boot (Java), which handles authentication, user role management, code routing, and result processing. The system offers customized test case management, real-time performance tracking, and role-specific dashboards for administrators, evaluators, and candidates. Its layered architecture enhances maintainability, scalability, and security, while the modular design allows for easy future expansion, such as the inclusion of plagiarism

detection or contest modes.

This platform is ideal for universities conducting coding labs, companies performing technical hiring, and organizations hosting programming contests. By combining the power of automation with robust design practices, this system delivers a flexible solution that bridges the gap between learning, practicing, and evaluating programming skills in a controlled, transparent, and scalable environment.

FIELD OF INVENTION:

The present invention relates to the field of **computer science and software engineering**, specifically in the areas of **automated code evaluation systems**, **online programming assessment platforms**, and **secure code execution environments**. It falls under the domain of **educational technology (EdTech)** and **technical recruitment automation**, where there is a growing demand for scalable, objective, and real-time evaluation of programming skills.

This invention addresses the challenges associated with manually evaluating code submissions by providing a **custom-built, web-based platform** that facilitates automatic compilation, execution, and result verification of user-submitted code. It supports a variety of programming languages and enables administrators and educators to define problem sets, test cases, and evaluation criteria. Furthermore, it incorporates essential components such as **sandboxing for secure code execution**, **RESTful API-based modular design**, and **role-based access control** to ensure safe and efficient operation.

The field of invention also includes technologies such as **containerization (Docker)** for environment isolation, **frontend development** using web technologies (HTML, CSS, JavaScript), **backend development** using Flask or Spring Boot, and **relational databases** like MySQL or PostgreSQL for structured data storage. These technologies are widely used in modern web applications, but their combined use in a customizable code evaluation platform offers a unique solution tailored for educational institutions, programming contests, and technical hiring processes.

This invention contributes to the field by providing a **flexible, extensible, and secure framework** that bridges the gap between academic evaluation and industry-standard coding assessments. It promotes efficiency, transparency, and accuracy in assessing programming capabilities, making it a significant development in the domain of automated educational and recruitment tools.

BACKGROUND OF INVENTION:

In recent years, the demand for skilled programmers has increased significantly across educational institutions, industries, and competitive environments. Traditionally, the

evaluation of programming assignments and coding tasks has been carried out manually by instructors or evaluators. While this approach offers a personal touch, it is highly time-consuming, subjective, and not scalable—especially in large-scale academic environments, online learning platforms, or during mass recruitment drives.

The advent of online learning and remote hiring has accelerated the need for **automated code evaluation platforms** that can assess coding skills accurately, quickly, and securely. These systems provide users with problem statements and allow them to submit code that is then automatically compiled, executed, and evaluated against predefined test cases. Several commercial and open-source platforms have emerged in this space, such as HackerRank, LeetCode, CodeRunner, and MOSS. However, these platforms often suffer from limitations such as lack of customization, vendor lock-in, limited integration capabilities, and the inability to operate in offline or institution-specific environments.

Moreover, executing user-submitted code introduces security risks, such as infinite loops, memory overflows, or malicious attempts to breach the system. Hence, there is a growing need for secure execution environments using containerization technologies like Docker, which can safely isolate and control the behavior of code running on shared infrastructure.

SUMMARY OF INVENTION:

The present invention relates to the development of a **secure, customizable, and scalable online code evaluation platform** that automates the assessment of programming assignments and technical coding challenges. This invention addresses the growing demand for rapid, unbiased, and secure code evaluation in academic institutions, recruitment processes, and competitive programming environments.

The platform is designed with a modular architecture that separates the user interface, application logic, execution engine, and database layers. It supports multi-language code execution—including C, C++, Java, and Python—within isolated Docker containers to ensure safe and consistent performance. The system facilitates automatic compilation, execution, and result verification against predefined test cases, offering real-time feedback and performance metrics to the users.

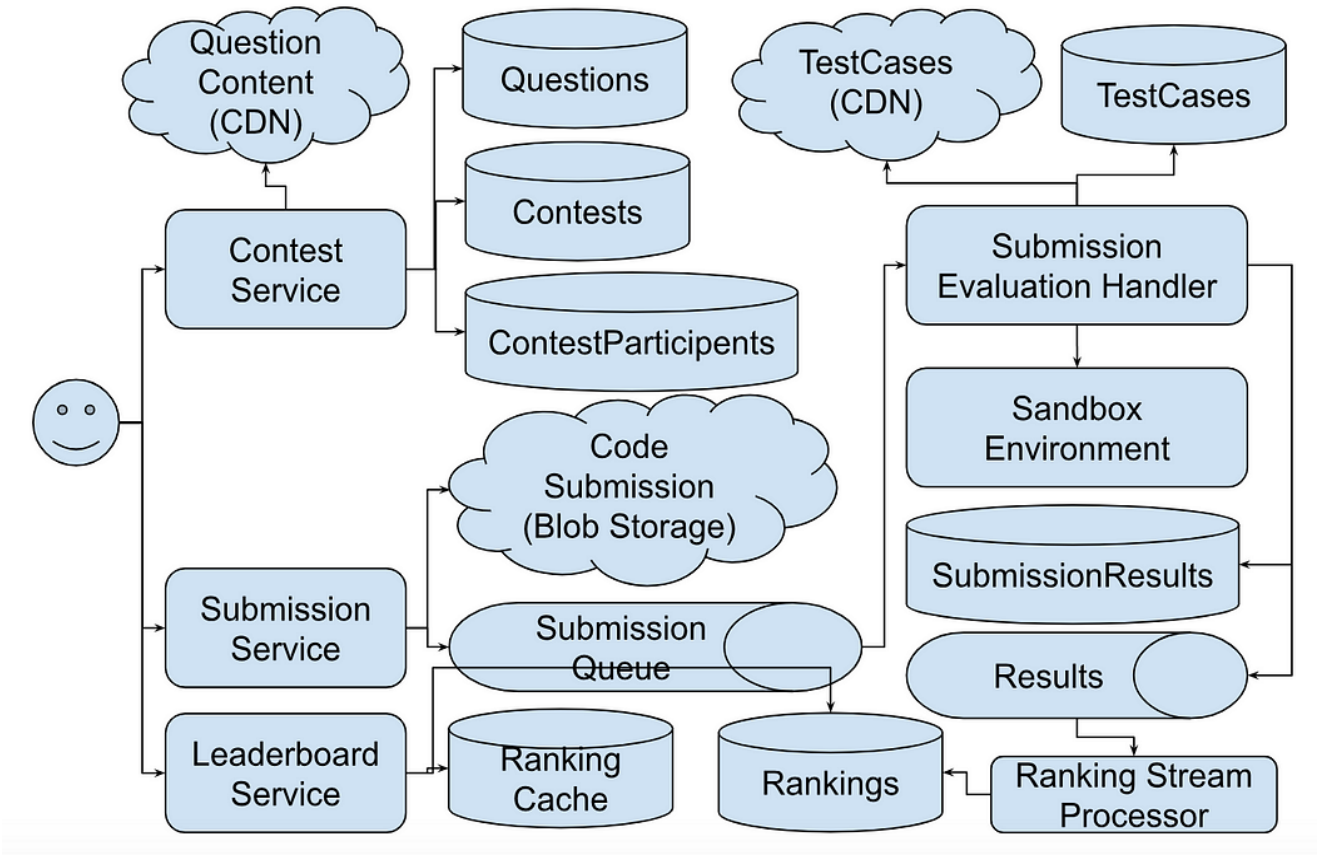
The invention includes functionalities for role-based access control, enabling administrators, evaluators, and students or candidates to interact with the system through customized workflows. Administrators can manage problems, test cases, and user roles; evaluators can monitor and manually review submissions if required; students can solve problems and view feedback instantly.

Additionally, the invention is built to be extensible. Its architecture supports the integration of future enhancements such as plagiarism detection tools, partial grading systems, and real-time contest environments. Security is ensured through container-based execution, input

validation, resource usage limits, and encrypted authentication mechanisms.

By overcoming the limitations of existing commercial platforms—such as lack of offline deployment, poor customization options, and integration difficulties—this invention provides a flexible and reliable solution tailored for educational and industrial use. It promotes efficient learning, fair evaluation, and smooth technical screening, thereby revolutionizing the way programming assessments are conducted.

DIAGRAM:



CLAIMS:

We claim that,

1. A modular architecture for an online code evaluation platform comprising a user interface, application logic layer, execution engine, and database, allowing clean separation of concerns and scalability.
2. A secure execution environment that utilizes Docker containers for sandboxed compilation and execution of user-submitted code, with enforced CPU, memory, and network restrictions.
3. A multi-language support mechanism that provisions language-specific containers to execute code in programming languages such as C, C++, Java, and Python.
4. A real-time code evaluation workflow that automatically compares submitted outputs with predefined test cases and provides instant feedback with execution metrics.
5. A role-based access control system where Administrators, Evaluators, and Students/Candidates interact through customized dashboards and features.
6. A secure authentication mechanism using token-based access (e.g., JWT) and password hashing techniques (e.g., bcrypt) to safeguard user credentials and sessions.