Marwadi University Marwadi Chandarana Group NAAC A+	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**	
Subject: Capstone Project (**01CT0715**)	**Aim**: Project Definition and Scope - Intermediate Review	
Project Documentation	**Date:**- 25-09-2025	**Enrollment No:-** 92200133030 92310133007

1.Introduction

- A successful career in software development requires more than just theoretical knowledge in the current technological environment. Tech companies actively test candidates for essential skills like algorithmic thinking and practical problem-solving during the hiring process. A common technique for developing these abilities is competitive programming, or CP. Students, however, frequently rely on a dispersed collection of open, non-curriculum-aligned online resources. This causes a gap between the demands of the industry and academic coursework.
- By creating a centralized, university-specific competitive programming platform, this project, called CodeArena, seeks to close this gap. Students will have a dedicated space on CodeArena to practice faculty-curated coding problems, such as those that were previously asked during on-campus recruitment drives. Instant feedback on submissions will be provided by the platform's secure remote code execution engine, user-friendly interface, and strong backend. The project's problem statement, goals, applicability, feasibility, and novelty are all described in this proposal.

2.Problem Statement

The fragmentation of learning resources for practical coding skills is the main issue that our university's computer science students face. This problem can appear in a number of ways:

- **Absence of a Centralized Hub:** Students currently utilize a variety of third-party platforms, such as LeetCode, HackerRank, and Codeforces, which are not specifically designed to meet the requirements of campus placement preparation or the university's curriculum.
- **Ineffective Faculty Content Management:** Instructors do not have a useful tool for uploading, organizing, and assigning pertinent coding problems. Informal, unstructured channels are frequently used to share issues from previous hiring campaigns.
- **Lack of Integrated Progress Tracking:** In a single system, it is challenging for teachers to keep track of students' progress, spot common problem areas, and offer focused support.
- **Theory and Industry Application Gap:** The theoretical ideas that are taught in the classroom and the real-world, demanding coding tests that employers use to hire

candidates differ greatly. For students to successfully mimic these technical interviews, a sandbox setting is necessary.

Thus, the primary issue is the lack of an integrated, university-run platform that synchronizes real-world coding practice with academic programs and industry standards, impeding the best possible development of students' problem-solving skills.

3.Objectives (SMART)

To address the problem statement, this project will achieve the following SMART objectives:

1. **Develop a Full-Featured Web Application:** To design and build a secure, scalable, and responsive web platform named "CodeArena" using React for the frontend and Laravel (PHP) for the backend within a 6-month timeframe. The platform will support distinct user roles (Student, Faculty) with corresponding functionalities.

2. **Integrate a Secure Remote Code Execution (RCE) Engine:** To implement and configure a self-hosted, containerized instance of **Judge0** to securely compile and execute user-submitted code in multiple programming languages (C++, Java, Python), ensuring an average evaluation and feedback time of less than 5 seconds per submission.

3. **Implement a Dynamic Problem Repository:** To create a robust system allowing faculty members to upload, edit, and categorize coding problems. Problems will be tagged by difficulty (Easy, Medium, Hard), topic (e.g., Arrays, Dynamic Programming), and company of origin, with a target of hosting at least 150 diverse problems by the project's conclusion.

4. **Enable Student Practice and Performance Analytics:** To provide students with an intuitive interface to solve problems, view their submission history, and track their performance. The system will maintain statistics on solved problems and submission accuracy.

4.Relevance to ICT Domain

CodeArena is deeply rooted in several key areas of the ICT domain:

- **Software Development:** The project encompasses the entire Software Development Life Cycle (SDLC), from requirement analysis and UI/UX design to implementation and testing. It utilizes modern web development frameworks (**React**, **Laravel**) and follows best practices for creating robust, maintainable applications.

- **DevOps and System Administration:** The deployment of a self-hosted, containerized Judge0 RCE engine is a direct application of DevOps principles. It involves using containerization technology (e.g., Docker) to ensure process isolation, security, and scalability, which are core challenges in modern system administration.
- **Cloud Computing & Microservices Architecture:** The architecture separates the main web application (monolith) from the code execution service (microservice). This modular design is a fundamental concept in cloud-native applications, allowing the RCE service to be scaled independently and ensuring that a failure in one component does not crash the entire system.
- **Educational Technology (EdTech):** The project is a direct contribution to the EdTech field, leveraging technology to create a tailored and enhanced learning experience for students, bridging the gap between academic education and professional skill requirements.

5.Feasibility Analysis

5.1 Technical Feasibility

The project is technically feasible as it relies on well-documented, open-source technologies.

- **Frontend: React** is a leading JavaScript library for building dynamic user interfaces, with extensive community support.
- **Backend: Laravel (PHP)** is a powerful, mature framework with built-in features for security, routing, and database management, making it ideal for rapid API development.
- **RCE Engine: Judge0** is a proven, open-source remote code execution system designed specifically for this type of application. Its containerized nature inherently provides a high degree of security and isolation. The project team possesses the requisite skills in these technologies to successfully complete the development.

5.2 Economic Feasibility

The project is highly cost-effective.

- **Software Costs:** All core technologies (React, Laravel, Judge0, PHP, MySQL) are open-source and free to use.
- **Hosting Costs:** Deployment can be done on a Virtual Private Server (VPS) provided by the university.

5.3 Ethical Considerations

- **Code Plagiarism:** The platform is for practice, but academic integrity is crucial. As a mitigation strategy, faculty will have access to submission timestamps and code. Future enhancements could include integrating a plagiarism detection tool.
- **System Security:** The RCE engine poses the largest security risk. By using a containerized Judge0 instance, any potentially malicious code is executed in an isolated environment, preventing it from accessing the host server or other users' data.

6. Market/User Needs Analysis

The primary users of CodeArena are **students** and **faculty** of the Marwadi University.

- **Student Needs:** Students require a reliable, centralized platform with a clean UI, instant feedback on their code, and a curated set of problems that are relevant to their curriculum and future job prospects.
- **Faculty Needs:** Faculty members need a simple, effective way to share practice material, create assignments, and monitor the practical skills development of their students without relying on external, ad-filled websites.

7.Literature Review & Novelty

A review of existing solutions reveals two main categories: large-scale commercial platforms and open-source judging systems.

- **Commercial Platforms (e.g., LeetCode, HackerRank):** These platforms are feature-rich but are generic by nature. They serve a global audience and lack the ability to be customized with curriculum-specific or university-centric content. Their business model is often based on recruitment services, which is beyond the scope of this project.
- **Open-Source Judges (e.g., DOMjudge, Kattis):** These are powerful systems, often used to host official programming contests. However, they are primarily backend engines and often lack a user-friendly, integrated learning management interface suitable for daily practice by students.

The **novelty of CodeArena** lies not in inventing a new algorithm for code execution, but in the **synthesis and customization of existing technologies to serve a specific institutional**

need. It combines a modern, user-friendly frontend (inspired by commercial platforms) with a secure, self-hosted backend (Judge0) and populates it with content that is directly relevant to the students and faculty of Marwadi University. Its unique value proposition is its role as a bespoke educational tool, fully controlled and curated by the university.

8. Conclusion

The CodeArena project proposes the development of a much-needed integrated competitive programming platform to enhance the practical problem-solving skills of students. By providing a centralized, secure, and curated environment, the platform directly addresses the current challenges of fragmented resources and the gap between academic theory and industry practice. The project is technically and economically feasible, leveraging modern open-source technologies. Upon successful completion, CodeArena will be a valuable asset to the university's ICT ecosystem, better preparing students for successful careers in the tech industry.