

# “Examination Portal for Coding, Aptitude and Reasoning Assessment”

Guide

**Prof. V. B. Soneji**

Author

**1. Rushikesh Kankale**

**2. Pranav Rase**

**3. Sanket Pachpol**

**4. Pratik Bhatkar**

**5. Shreyash Deshmukh**

## ABSTRACT

The Examination Portal (EP) is an integrated online assessment platform designed to conduct coding, aptitude, and reasoning tests within a single centralized environment. It combines candidate registration, test scheduling, question-bank management, a secure code-execution engine, automated aptitude and reasoning evaluation, proctoring, result generation, and analytics into one unified system. The system uses centralized cloud hosting, authentication, and Role-Based Access Control (RBAC) to provide secure and role-specific access to stakeholders. Developed using a structured approach involving system design, frontend and backend implementation, API integration, sandboxed code execution, security, testing, and deployment, EP aims to improve assessment standardization, evaluation accuracy, scalability, and the overall candidate and administrator experience.

*Keywords: Online Examination, Coding Assessment, Aptitude Testing, Reasoning Assessment, Web-Based Portal, Auto-Evaluation, Proctoring, Role-Based Access Control.*

## 1. INTRODUCTION

### 1.1 Background

Recruitment drives, academic evaluations, and skill-certification programs increasingly rely on structured assessments that combine coding ability with aptitude and logical reasoning. Conducting these three assessment types through separate tools, paper-based tests, or disconnected platforms often leads to inconsistent evaluation standards, manual result compilation, scheduling conflicts, and difficulty in detecting malpractice. This creates the need for a single, unified examination portal capable of handling all three assessment categories under one consistent framework. [10], [13]

### 1.2 Web-Based and Automated Assessment

A web-based examination portal enables candidates to attempt tests remotely while allowing administrators to manage question banks, schedules, and evaluation centrally. Automated evaluation, through a sandboxed code-execution engine for programming questions and rule-based scoring for aptitude and reasoning questions, reduces manual effort, minimizes human error, and accelerates result declaration.

Since candidate data, test content, submissions, and results are interconnected, integrating these processes within a unified portal improves consistency, transparency, and administrative efficiency. [1], [2], [4], [6]

### 1.3 Existing Challenges [3], [7], [16]

Disconnected assessment practices may cause inconsistent question standards across coding, aptitude, and reasoning sections, delayed or manual evaluation of code submissions, limited scalability for large candidate pools, inadequate proctoring and malpractice detection, and fragmented reporting of results. These challenges highlight the requirement for an integrated, automated, and secure examination environment.

### 1.4 Proposed Examination Portal

The Examination Portal (EP) is proposed as a centralized platform integrating candidate registration, test and question-bank management, coding assessment with real-time code execution, aptitude and reasoning assessment with auto-evaluation, proctoring, result processing, dashboards, and analytics. [2], [9], [10], [12]

Workflow:

Candidate Registration → Test Enrollment → Coding / Aptitude / Reasoning Modules → Submission → Auto-Evaluation → Proctoring Review → Result Generation → Dashboard & Analytics

The system incorporates authentication and Role-Based Access Control (RBAC) to provide secure, role-specific access to candidates, examiners, and administrators. Overall, EP aims to improve assessment standardization, evaluation accuracy, scalability, and stakeholder experience throughout the examination lifecycle. [8], [18]

## 2. PROBLEM STATEMENT

Conducting coding, aptitude, and reasoning assessments involves interconnected activities including candidate management, question-bank curation, test scheduling, secure code execution, automated scoring, malpractice detection, and result reporting. Managing these activities through separate or manual tools can cause inconsistent evaluation standards, delayed result processing, limited scalability, and difficulty in monitoring test integrity. [1], [2], [3], [4], [5], [8]

Existing digital solutions often address individual assessment types, either coding judges, generic quiz platforms, or standalone proctoring tools, rather than providing an integrated workflow for combined coding, aptitude, and reasoning evaluation. There is therefore a need to connect candidate data, question banks, submissions, evaluation, and results within a single environment, while also ensuring secure and role-based access. [1], [2], [3], [12], [15]

To address these challenges, the Examination Portal (EP) is proposed as a centralized web platform. The system aims to provide standardized assessment delivery, automated and accurate evaluation, secure access, scalable test hosting, and integrated result analytics, thereby reducing dependency on disconnected and manual examination processes. [5], [9], [10]

### 3. OBJECTIVES

The primary objective of this research is to develop a Examination Portal (EP) that provides a centralized, automated, and secure environment for conducting coding, aptitude, and reasoning assessments.

The specific objectives are:

1. To develop a centralized web-based platform for conducting coding, aptitude, and reasoning assessments. [2], [9], [10]
2. To integrate major functional modules including candidate management, question-bank management, test scheduling, code execution, auto-evaluation, proctoring, and reporting. [1], [2], [3], [5], [9]
3. To implement a sandboxed code-execution engine capable of compiling and evaluating candidate submissions in multiple programming languages. [1], [4], [12], [15], [16]
4. To establish an automated evaluation mechanism for aptitude and reasoning sections that minimizes manual scoring effort and delay. [2], [6]
5. To provide centralized dashboards and analytics for monitoring candidate performance, test statistics, and evaluation trends. [5], [10], [16]
6. To implement secure authentication, proctoring, and Role-Based Access Control (RBAC) for controlled and tamper-resistant access to system resources. [3], [7], [8], [18]
7. To evaluate the system in terms of functionality, evaluation accuracy, usability, security, scalability, and effectiveness in supporting examination management.

### 4. LITERATURE REVIEW

Prior work on online judging platforms has focused mainly on automated compilation and test-case-based evaluation of programming submissions, but such systems typically remain limited to coding assessment without extending to aptitude or reasoning evaluation. Studies on e-learning and

online examination systems have examined question randomization, timed delivery, and result automation for objective-type tests, though these approaches are largely oriented toward academic quizzes rather than integrated recruitment-style assessments. Research on remote proctoring has explored webcam- and browser-based monitoring techniques to reduce malpractice in unsupervised test environments, but integration of such proctoring with multi-domain assessment together remains limited. Work on cloud-based scalable assessment infrastructure has addressed handling large concurrent candidate loads during examination windows, but often without a unified data model connecting candidate profiles, submissions, and analytics. [1], [12], [15], [16], [17]

Overall, existing approaches support different aspects of examination management, coding judges, quiz engines, proctoring tools, or scalable hosting, but lack a single, lightweight platform that unifies coding, aptitude, and reasoning assessment with automated evaluation and analytics. EP addresses this gap by integrating candidate management, question banks, code execution, auto-evaluation, proctoring, and reporting within a unified web-based platform. [1], [2], [3], [5], [9], [10], [12], [15]

### 5. PROPOSED METHODOLOGY

#### 5.1 Proposed System Overview

EP is a centralized web-based platform for conducting and managing coding, aptitude, and reasoning assessments. It integrates candidate, question-bank, test, submission, evaluation, proctoring, and reporting activities, reducing manual effort and ensuring evaluation consistency through secure, role-based access. [2], [9], [10], [12]

Workflow:

Candidate Registration → Test Enrollment → Question Delivery (Coding / Aptitude / Reasoning) → Submission → Auto-Evaluation → Proctoring Review → Result Compilation → Dashboard & Reports

#### 5.2 System Architecture

EP consists of four layers: the User Layer for role-based access by candidates, examiners, and administrators; the Application Layer for system modules including the test engine and code-execution service; the Database Layer for centralized storage of candidates, questions, submissions, and results; and the Cloud & Security Layer for hosting, sandboxing, and access security. [5], [8], [18]

Architecture Flow:

Users → Application Interface → EP Modules (Test Engine, Code Execution Sandbox, Evaluation Engine) → Database → Cloud Services

This architecture supports scalable concurrent test delivery, isolated and secure code execution, and controlled access across candidate, examiner, and administrator roles. [4], [5], [8], [16]

### 5.3 System Modules

EP comprises nine modules covering authentication, candidate registration, test and question-bank management, coding assessment with sandboxed code execution, aptitude and reasoning assessment with auto-evaluation, timed test delivery, proctoring, result processing, and dashboards and reporting. Together, these modules support the complete examination lifecycle from candidate onboarding to result declaration and performance analytics. [1], [2], [3], [4], [6], [9], [12]

#### Overall Workflow

Candidate → Registration → Test Assignment → Coding / Aptitude / Reasoning Modules → Submission → Evaluation → Proctoring → Result → Dashboard → Reports → Analytics and Trends

## 6. RESULTS AND DISCUSSION

### 6.1 Results

The development of EP has established a centralized platform for conducting coding, aptitude, and reasoning assessments. The implemented Authentication and Test-Delivery modules provide secure login, role-based access, question rendering, timed test sessions, and submission handling. The system is integrated using a modern web stack with REST APIs, a relational database for structured data storage, and a sandboxed execution service for compiling and running candidate code submissions, providing structured data processing and centralized result storage. [1], [2], [12], [15]

### 6.2 Discussion

The results indicate that the modular architecture provides a scalable foundation for further development. Integration between test delivery and the code-execution engine improves evaluation consistency and reduces manual scoring effort, while authentication and role-based access ensure controlled system usage. Further evaluation will be conducted after integrating and testing the remaining modules, including aptitude and reasoning auto-evaluation, proctoring, result analytics, and administrative reporting dashboards. [4], [5], [8], [12], [16]

## 7. LIMITATIONS

The current development of EP has certain limitations related to implementation scope, evaluation, cloud dependency, access control, and scalability.

### 7.1 Partial Implementation

The current system mainly implements the Authentication and Test-Delivery modules, while the remaining modules, including proctoring and full analytics, are under progressive

development. Hence, complete system functionality cannot yet be evaluated.

### 7.2 Limited Evaluation

Testing is currently focused on implemented modules. System-wide performance, evaluation accuracy, usability, and end-to-end workflow require evaluation after complete integration.

### 7.3 Cloud and Network Dependency

EP relies on cloud-hosted infrastructure and REST-based communication for test delivery and code execution; therefore, system availability and evaluation operations depend on reliable network and cloud services. [5]

### 7.4 Access-Control Expansion

The existing role-based access control may require further refinement to support module-specific permissions and workflows as additional modules, such as proctoring and analytics, are integrated. [8], [18]

### 7.5 Scalability and Performance

Large-scale performance under high concurrent candidate load, particularly during simultaneous code execution, has not yet been fully evaluated. Future testing will require load testing, sandbox-resource optimization, and scalability assessment. [5], [16]

### 7.6 Overall Limitation

These limitations reflect the current development stage. Completion and integration of the remaining modules, followed by comprehensive testing, will enable a complete evaluation of EP in terms of functionality, security, evaluation accuracy, performance, and scalability.

## 8. References

- [1] A. Kumar and S. Sharma, "Design and Implementation of an Automated Online Judge System for Programming Assessment," *International Journal of Computer Applications*, vol. 178, no. 30, pp. 1–6, 2019.
- [2] R. Singh and P. Verma, "A Web-Based Online Examination System with Automated Evaluation," *International Journal of Engineering Research and Technology*, vol. 9, no. 4, pp. 112–117, 2020.
- [3] M. Petrovic and D. Simic, "Remote Proctoring Techniques for Online Assessment: A Survey," *Education and Information Technologies*, vol. 26, pp. 6421–6445, 2021.
- [4] T. Nguyen, H. Le, and V. Tran, "Sandboxed Code Execution Frameworks for Competitive Programming Judges," *Journal of Systems and Software*, vol. 178, Art. no. 110981, 2021.
- [5] S. Gupta and N. Malhotra, "Cloud-Based Scalable Architecture for Large-Scale Online Examinations,"

- Procedia Computer Science, Elsevier B.V., vol. 218, pp. 1452–1460, 2023.
- [6] K. Patel, J. Shah, and R. Mehta, "Automated Aptitude and Reasoning Test Evaluation Using Rule-Based Scoring," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 6, pp. 210–216, 2021.
- [7] L. Chen and Y. Wang, "A Review of AI-Based Proctoring and Academic Integrity Systems in Online Learning," *Computers & Education*, vol. 189, Art. no. 104580, 2022.
- [8] F. Rahman et al., "Role-Based Access Control Models for Secure Web Applications: A Systematic Review," *Journal of Network and Computer Applications*, vol. 202, Art. no. 103375, 2022.
- [9] Prajapati, H., Patel, V., and Bhavsar, J., "Development of an Integrated Web-Based Assessment Module for Academic Institutions," *Innovative Infrastructure Solutions*, Springer Nature, vol. 7, no. 2, pp. 88–101, 2022.
- [10] Ullah, K., Nagpal, A., and Kumar, A., "Applications of Automated Assessment Systems for Enhancing Recruitment and Academic Evaluation: A Review," *Heliyon*, Elsevier, vol. 9, no. 3, 2023.
- [11] "Design and realization of an online examination system based on Maven framework," in *Proc. 3rd Int. Conf. Computer Science and Management Technology (ICCSMT)*
- [12] Mekterovic, L. Brkic, B. Milasinovic, and M. Baranovic, "Building a comprehensive automated programming assessment system," *IEEE Access*, vol. 8, pp. 81154–81172, 2020
- [13] M. Farrow and P. J. B. King, "Experiences with online programming examinations," *IEEE Trans. Educ.*, vol. 51, no. 2, pp. 251–255, May 2008
- [14] P. E. Robinson and J. Carroll, "An online system for monitoring and assessing the programming process," *IEEE Computer Society*, 2016.
- [15] R. Lobb and J. Harlow, "CodeRunner: A tool for assessing computer programming skills," *ACM Inroads*, vol. 7, no. 1, pp. 47–51, 2016
- [16] I. Mekterovic, L. Brkic, and M. Horvat, "Scaling automated programming assessment systems," *Electronics*, vol. 12, no. 4, p. 942, 2023
- [17] N.-L. Hsueh, L.-C. Lai, and W.-H. Tseng, "Design of an online programming platform and a study on learners' testing ability," *Electronics*, vol. 12, no. 22, p. 4596, 2023
- [18] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-based access control models," *Computer*, vol. 29, no. 2, pp. 38–47, Feb. 1996.