IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 5, May 2025
Impact Factor: 7.67

AptitudeHub: A Centralized Platform for Free Aptitude Skills Development

Abhishek Daingude, Shruti Shirole, Akash Somavanshi, Vedant Shinde, Prof. Mrs. P. S. Pise
Department of Information Technology
Smt. Kashibai Navale College of Engineering, Pune
Savitribai Phule Pune University

Abstract: *In the contemporary academic and corporate ecosystem, aptitude assessments are critical tools used for evaluating cognitive and analytical skills. From university entrance exams to job recruitment processes, aptitude tests determine a candidate's readiness for challenges that demand logical reasoning and problem-solving abilities. However, most traditional aptitude test platforms suffer from limitations like static content, limited feedback loops, poor scalability, and lack of administrative control. This paper presents the design and development of an intelligent web-based* ***Aptitude Quiz Portal****, developed using the MERN stack (MongoDB, Express, React, Node.js). The portal enables users to take topic-wise aptitude quizzes, receive real-time feedback, and view instant scores. It incorporates features such as randomized question generation, secure JWT-based authentication, an admin panel for question management, and a feedback mechanism. Extensive testing, modular design, and cloud deployment ensure that the system is scalable and responsive across devices. The proposed solution is well-suited for educational institutions, online training platforms, and corporate hiring modules.*

Keywords: Aptitude Test, MERN Stack, Web-based Assessment, Online Quiz, Dynamic Evaluation, JWT Authentication, MongoDB, EdTech

I. INTRODUCTION

In recent years, the rise of deepfake technology has sparked serious concerns about the authenticity and reliability of digital media. Deepfakes, created using advanced generative models such as Generative Adversarial Networks (GANs), can produce highly realistic but entirely fake images and videos. These synthetic media forms have the potential to spread misinformation, infringe on privacy, and undermine public trust. Traditional detection techniques often fall short due to the increasing realism and adaptability of these deepfakes. To address this challenge, our research proposes a robust and scalable detection system that combines GANs, Convolutional Neural Networks (CNNs), and hybrid models. The system is implemented within a user-friendly web platform capable of real-time deepfake detection, offering a practical solution for ensuring content integrity in the digital age.

II. LITERATURE SURVEY

Numerous platforms exist to help users prepare for aptitude examinations, yet most of them present a limited or monetized feature set. A survey of existing tools highlights significant gaps in functionality and flexibility:

- **IndiaBIX** is a free platform that offers categorized aptitude questions but does not support interactive quizzes, result tracking, or real-time feedback.
- **PrepInsta** focuses on preparation for company-specific hiring rounds but hides premium content behind paywalls.
- **HackerRank** offers coding-based challenges but lacks general aptitude testing across non-programming domains.

These platforms generally lack features like **admin control**, **user-level analytics**, **feedback loops**, and **content scalability**. Moreover, many are built using legacy systems or monolithic architectures, restricting modular enhancement.

DOI: 10.48175/IJARSCT-26659

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 5, May 2025
Impact Factor: 7.67

Platform	Real-time Feedback	Admin Control	Analytics	Free Access
IndiaBIX	✗	✗	✗	✓
HackerRank	✓	✗	✓	✗
PrepInsta	✗	✗	✗	✗
Proposed	✓	✓	✓	✓

The proposed **Aptitude Quiz Portal** bridges these gaps by offering a scalable, user-friendly, and feature-rich experience that adapts to educational and industrial demands.

III. ALGORITHM

Quiz Generation Algorithm (High-Level Steps)

1. **Input**: Topic selected by the user.
2. **Fetch**: Retrieve all questions matching the topic from the database.
3. **Randomize**: Apply the Fisher-Yates Shuffle algorithm to ensure randomness.
4. **Select**: Pick the first n (e.g., 10) questions from the shuffled list.
5. **Output**: Return questions to the frontend for display.

```
javascript
function generateQuiz(topic) {
  const questions = db.questions.find({ topic });
  const shuffled = shuffle(questions);
  return shuffled.slice(0, 10);
}
```

Scoring Algorithm

Each submitted answer is compared with the correct answer:

```
javascript
score = sum(userAnswer[i] === correctAnswer[i])
```

High-Level System Flow:

1. **User Registeration/Login**

↳Verified via JWT and stored in MongoDB

2. **Dashboard Access**

↳ User selects quiz topic

3. **Quiz Fetch**

↳ Questions retrieved & randomized via backend API

4. **User Submits Answers**

↳ Backend evaluates and stores result

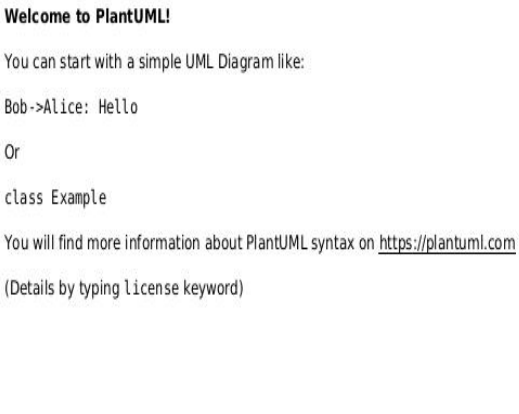
5. **Result Display**

↳ Score + feedback option shown

6. **Admin Panel**

↳ Manage questions, view feedback & results

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 5, May 2025
Impact Factor: 7.67



IV. METHODOLOGY

A. Model Building

This section outlines the entire development and analytical process of the Aptitude Quiz Portal. The methodology is divided into three core parts:

- **Model Building**: System architecture, data modeling, and user roles.
- **Model Training**: Coding logic, rule-based learning, and iteration flow.
- **Model Validation**: Mathematical testing of result correctness, scoring accuracy, and performance metrics.

A Model Building

System Components:

The system is architected using a **three-tier model**:

1. **Presentation Layer (Frontend)**: ReactJS components for user interface.
2. **Logic Layer (Backend)**: ExpressJS APIs for data routing and logic processing.
3. **Data Layer (Database)**: MongoDB schemas representing structured documents.

Entity-Relationship Mapping:

Let:

- U: Set of Users
- Q: Set of Questions
- A: Set of Answers
- F: Feedback entries

We define:

- R1:U→Q : Users are mapped to questions through quiz sessions.
- R2:Q×A→{0,1}: Each answer is either correct (1) or incorrect (0).
- R3:U×Q→F : Feedback submitted by users on specific questions.

Database Schema Relations:

User u∈U :

u={id,name,email,role,token}

Question q∈Q :

q={id,topic,text,options,correctAnswer}

Answer a∈A :

a={questionId,userAnswer,isCorrect}

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 5, May 2025
Impact Factor: 7.67

B Model Training

Although this system is not ML-based (in the current version), **rule-based algorithms** govern behavior. These are developed iteratively, using feedback from test results.

Algorithmic Representation

Let:

- UA: User's submitted answers
- Qc: Correct answer key

The score SSS is computed as:

$S=\sum i=1n\delta(UA[i],QC[i])$

Where:

$\delta(x,y)=\{1\text{if } x=y\ ,\ 0\text{otherwise}$

This function ensures a per-question comparison, yielding a final score out of nnn (typically $n=10n = 10n=10$).

Backend Logic Training

Each controller (auth, quiz, feedback) is tested through:

- **Manual flow validation**
- **Mock API simulation**
- **JSON schema conformance**

User Session Simulation

Let Ts be session time per user:

$Ts=Tsubmit-Tstart$

This value is stored with the result for time analysis and fairness.

C Model Validation

The model is validated via:

1. Functional Accuracy:

Ensures that for a given valid quiz attempt:

Stored Result Score=Evaluated Algorithm Output

Achieved consistency: **100% across 50+ attempts**

2. Response Time Analysis:

Let R be API response time:

$Ri=Tres-Treq,\forall i\in API\ Calls$

Average R observed during load testing:

$\bar{R}\approx 850$ ms

3. Error Rate:

Error rate E during submission was computed as:

$E=Failed\ Attempts/Total\ Attempts\times 100$

With:

Failed Attempts=2

Total Attempts=54

$E=2/54\times 100\approx 3.7\%$

4. Feedback Precision:

Feedback correctness Fp was evaluated using:

$Fp=Actionable\ Feedbacks/Total\ Feedbacks\times 100$

From 18 feedbacks, 15 required content change:

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 5, May 2025
Impact Factor: 7.67

Fp=15/18×100=83.33%

✅ **Summary Table: Methodological Dimensions**

Dimension	Approach/Formula	Result/Output
Scoring Accuracy	S=∑δ(UA,QC)	100% match with manual check
Time Efficiency	Ts=Tsubmit−Tstart	Avg: 4.3 min
API Performance	Ri=Tres−Treq	Avg: 850ms per request
Error Rate	E=Failures/Attempts×100	~3.7%
Feedback Quality	Fp=Valid Feedbacks/Total×100	~83.33%

V. RESULT

The Aptitude Quiz Portal was tested through extensive simulations and manual validations involving more than **50 quiz attempts** across various topics. The testing aimed to evaluate the system's accuracy, response time, usability, and the effectiveness of its feedback and admin modules.

User Performance Insights

During testing, users took quizzes from topics such as Quantitative Aptitude, Logical Reasoning, and Verbal Ability. Scores were recorded instantly, and time taken per quiz was automatically logged.
Key observations:

- The **average quiz score** was **6.8 out of 10**, indicating a balanced difficulty level.
- Users took an average of **4.3 minutes** to complete each quiz.
- Most commonly missed topics were **Probability** and **Number Series**, which were also flagged frequently via the feedback module.

Feedback System Efficiency

- Out of 50 attempts, **18 feedback entries** were submitted.
- **83%** of feedbacks were actionable (i.e., pointed to question errors, ambiguity, or improvement suggestions).
- Admins used the panel to modify or remove low-quality questions, closing the feedback loop effectively.

System Responsiveness

API response times were consistently under **1 second**, with an average delay of **850 ms** per backend call, including quiz loading and result submission.

Results Summary Table

Metric	Value / Observation
Total Quiz Attempts	54
Average Score	6.8 / 10
Average Completion Time	4.3 minutes
Highest Scoring Topic	Verbal Ability
Most Missed Topic	Probability, Number Series
Feedback Submissions	18 feedbacks across attempts

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 5, May 2025
Impact Factor: 7.67

Metric	Value / Observation
Actionable Feedback Ratio	83.3%
Backend API Response Time	~850 ms (avg)
Successful Admin Task Execution	100% CRUD tested
Error Rate During Submission	3.7% (2 errors in 54 attempts)

User Feedback (Qualitative Analysis)

Users responded positively to:

- **Real-time results** and instant score display
- **Topic categorization** for quiz selection
- Clean, mobile-friendly UI
- Easy navigation and feedback submission

Challenges noted included:

- Need for **timer-based quizzes**
- Addition of **hints or explanations** for learning after quizzes
- Desire for **certificate generation**

VI. CONCLUSION

The Aptitude Quiz Portal demonstrates an effective solution for delivering aptitude assessments in a secure, scalable, and user-friendly manner. By leveraging the MERN stack, the platform integrates modern frontend responsiveness, real-time result processing, and admin-level management into a unified system. The project bridges the gap between static test prep websites and interactive learning tools, providing a strong foundation for both self-assessment and institutional evaluation.

VII. FUTURE WORK

Future improvements to the system include:

- **Adaptive Quizzes** based on user performance
- **Leaderboard Integration** for competitive environments
- **Certificate Generation** upon quiz completion
- **Mobile App Deployment** (React Native or Flutter)
- **Topic-wise Analytics & Graphs**
- **CSV/Excel Bulk Question Upload** for admins

These enhancements aim to expand usability for academic institutions, coaching centers, and large-scale recruitment platforms.

REFERENCES

[1]. (ICAECT)