



## **BANNARI AMMAN INSTITUTE OF TECHNOLOGY**

An Autonomous Institution Affiliated to Anna University - Chennai, Accredited by NAAC with A+ Grade

Sathyamangalam - 638401 Erode District, Tamil Nadu, India

<b>STUDENT NAME</b>	<b>PRIYALAKSHMI N</b>
<b>ROLL NUMBER</b>	<b>7376221CS270</b>
<b>SEAT NO</b>	<b>52</b>
<b>PROJECT ID</b>	<b>01</b>
<b>PROJECT TITLE</b>	<b>Assessment Platform</b>

### **1. INTRODUCTION:**

Assessment software allows educators to prepare and give examinations to students using digital devices such as PCs, cellphones, and tablets. These systems simplify the process of administering tests, evaluating student responses, and reporting results. This test is carried out via the Internet utilizing existing web technologies. An evaluation may have defined objectives, such as testing an individual's expertise or learning.

## **1.1 PURPOSE:**

The fundamental purpose of assessment is to gain an in-depth knowledge of what students know, learn, and can accomplish with that knowledge. It supports students in demonstrating their learning, providing feedback on errors they've made, and providing opportunities to improve their performance with each assessment.

## **1.2 SCOPE:**

The main objectives of an assessment platform are to create and manage assessments, provide and administer tests, engage learners with feedback, automate grading and reporting, integrate with other systems, ensure security and compliance, and provide scalability and flexibility. It allows instructors to create exams, students to complete them, and administrators to monitor performance and learning outcomes throughout an organization.

## **2.SYSTEM OVERVIEW:**

### **2.1 Admin Roles:**

**1. Platform Administrator:** Responsible for overall platform management, including user accounts, access controls, system configurations, and integrations.

**2. Assessment Manager:** Defines assessment criteria, templates, and rubrics. Manages the assessment workflow and reporting.

**3. Integration Specialist:** Configures connections to other enterprise systems for data exchange and automation.

### **2.1.1 Admin Features:**

**User management:** Create, update, and disable user accounts with appropriate permissions.

**Access control:** Define and manage role-based access to platform features and data.

**Assessment configuration:** Set up assessment templates, criteria, and scoring rubrics.

**Integration management:** Configure integrations with other systems (e.g., student information, LMS).

**System settings:** Customize platform branding, notifications, and preferences.

**Audit and monitoring:** Track user activities, system performance, and security events.

### **2.2 Student Roles:**

**1. Student:** Participate in projects, complete assigned tasks, and contribute to assessments.

**2. Project Lead:** Responsible for managing a project team, delegating tasks, and coordinating with the instructor.

#### **2.2.1 Student Features:**

**Task management:** View assigned tasks, update progress, and submit deliverables.

**Assessment participation:** Contribute to project assessments, provide feedback, and respond to instructor comments.

**Booking assessment slots:**

- View a calendar of available assessment timeslots.
- Select an open slot and reserve it for their project assessment.
- Receive a confirmation once a slot is booked, and the slot is marked as unavailable for other students.
- View their scheduled assessment slot and any instructions or requirements.
- Get reminders from the platform about their upcoming assessment slot.
- Cancel their existing booking and select a new available slot if they need to reschedule.

**Notifications:** Receive alerts for upcoming deadlines, assessments, and important updates.

**Performance tracking:** View personalized dashboards and reports on project progress and assessment results.

**3. TECHNOLOGY STACK:**

Backend	Python(Flask or Django)
Database	PostgreSQL or MySQL
Frontend	React.js or Vue.js
Deployment	Docker and Kubernetes

**3.1 Backend Requirements:****User management:**

- Create, update, and delete user accounts
- Implement user roles (admin, student)

**Assessment management:**

- Define assessment templates and scoring rubrics
- Manage assessment workflow (submission, review, grading)
- Generate assessment reports

**Booking assessment slots:**

- Maintain a calendar of available assessment timeslots
- Allow students to reserve and manage their slots
- Send notifications to students about their scheduled slots

**Notification system:**

- Send alerts and reminders to users for deadlines, assessments, and updates

**3.2 Frontend Requirements:****User interface:**

- Responsive and user-friendly screens for admin and students
- Intuitive navigation and workflows

**Assessment participation:**

- Students contribute to assessments, provide feedback, and respond to instructors
- Display assessment details, instructions, and grading criteria

**Booking assessment slots:**

- Calendar view to show available time slots
- Interface for students to reserve and manage their slots
- Display scheduled assessment details and countdown timers

**Notification center:**

- Showcase alerts, reminders, and updates for user

### **3.3 Deployment and Infrastructure:**

#### **Containerization:**

- Package backend, frontend, and supporting services into Docker containers

#### **Orchestration:**

- Use Kubernetes to manage the deployment, scaling, and networking

#### **Automation:**

- Implement CI/CD pipelines for automated build, test, and deployment
- Configure monitoring and logging solutions

## **4. FUNCTIONAL REQUIREMENTS:**

### **User Roles and Permissions:**

#### **Admin:**

- Administrators should have full access to user management, assessment management, reporting, and platform configuration.

#### **Students:**

- Students should have access to assessment participation, booking slots, and viewing their own performance data.

### **Authentication Methods:**

#### **Admin:**

- Implement a traditional username and password-based authentication for administrators.

#### **Students:**

- Allow students to sign in using their Google accounts.

## **User Registration and Onboarding:**

### **Admin:**

- Allow administrators to create their own accounts with the platform.
- Require administrators to provide a valid email address and set a secure password.

### **Students:**

- The Google account integration to automatically create user accounts during the sign-in process.

## **User Management:**

### **Admin:**

- Allow administrators to create, update, and deactivate user accounts.
- Enable administrators to assign, modify, or revoke user roles and permissions.

### **Students:**

- To update their profile information, change passwords, or request account assistance.

## **Security and Compliance:**

- Use secure password hashing and salting
- Enforce strong password policies
- Regularly review and update security measures.

## **User Experience:**

- **Mobile-Friendly:** Users should be able to access the platform seamlessly on both desktop and mobile devices.
- **Clear Instructions:** Provide clear guidance on how to use the platform's features and functionalities.
- **Feedback:** Regularly gather user feedback and improve the platform over time.

## FLOWCHART:

