**DOCUMENTATION FOR FITNESS CERTIFICATE PORTAL**

|  |  |
| --- | --- |
| **NAME** | Nikithaa G |
| **ROLL NO** | 7376221CS244 |
| **SEAT NO** | 67 |
| **PROJECT ID** | 27 |
| **PROBLEM STATEMENT** | Fitness Certificate Portal |

**1. INTRODUCTION:**

**1.1. Purpose:**

This document outlines the concept of a Fitness Certificate (FC) portal designed for colleges. The portal's primary objective is to facilitate venue fitness checking, wherein college administrators can log in, provide updates about various venue categories through yes/no type questions, and receive alerts regarding non-compliance issues.

**2. SYSTEM OVERVIEW:**

**2.1. Users:**

* College Administrators: These users will have access to the portal to update venue fitness status regularly.
* Admins: System administrators overseeing the FC portal's functionality, managing submissions, and generating reports.

**2.2. Features**:

* Login and Registration:
* College administrators can register or log in to their accounts.
* Admins have privileged access for system management.
* Venue Fitness Updates:
* College admins update venue fitness by answering yes/no questions regarding different categories (e.g., facilities, safety measures).
* Regular updates are required, preferably bi-monthly.
* Non-compliance Alerts:
* The system tracks responses and flags instances where three consecutive "NO" responses occur within a subcategory.
* Admins receive notifications and can access lists of venues requiring attention.
* Analytics and Reporting:
* Admins can view overall fitness status, number of compliant venues, non-compliant venues, trends in non-compliance, etc.
* Detailed reports are available for decision-making and follow-up actions.

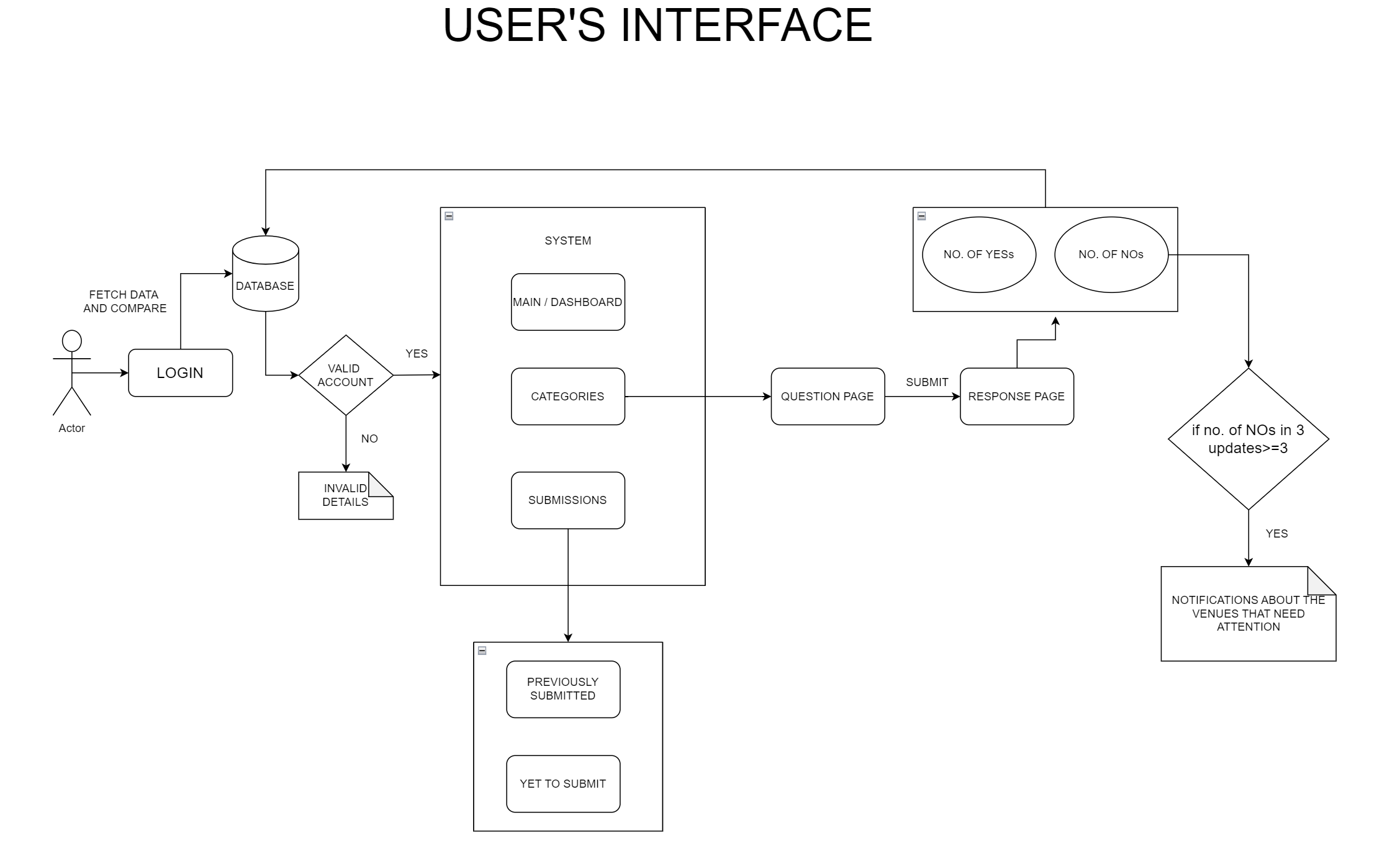
**2.3. Project Constraints:**

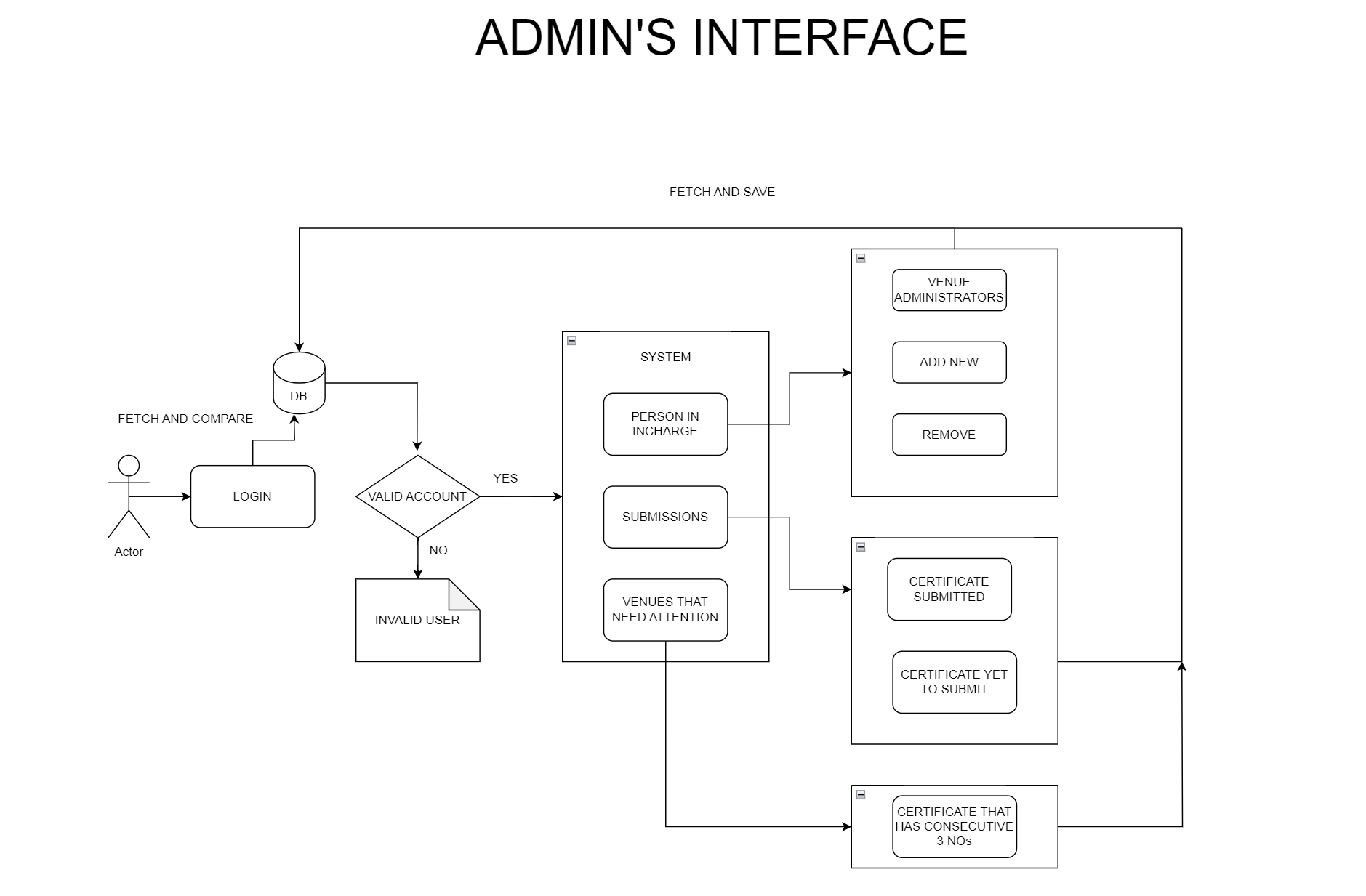
* Frequency of Updates: Administrators must update venue fitness status at least twice a month to ensure data accuracy.
* Alert Mechanism: The system must accurately identify and promptly notify admins about non-compliant venues based on set criteria.
* User Accessibility: The portal should be user-friendly for both college administrators and system admins, ensuring easy navigation and data input.
* Data Security: Robust measures should be in place to safeguard sensitive information within the portal.

**2.4 External Stimuli Reaction:**

* Non-Compliance Handling: Upon identifying non-compliant venues, the system generates immediate alerts for admins, highlighting specific areas of concern for corrective action.
* User Support: Help resources, FAQs, and user guides are available within the portal to assist administrators in navigating and utilizing the system effectively.
* Feedback Mechanism: The portal includes feedback mechanisms for users to report any technical issues or suggest improvements, enhancing overall user experience.

This Fitness Certificate portal aims to streamline venue fitness management within colleges, ensuring timely updates, proactive identification of non-compliance, and efficient communication between administrators and system managers.





**3. FUNCTIONAL REQUIREMENTS:**

* **User Management:** Students can register and log in. Admins have access control with an analytical dashboard and dedicated features.
* **TAC Application:** Students can submit applications with appropriate details. Application form contains Title of Project, Category of the project, Number of students involved, and Provisional document attachment.
* **Application Status:** Students can view the current status of their application. If the application is rejected, then the remarks are shown. Students can also see the logs of their applications.
* **Appointment Scheduling (After Approval):** Students with approved TACs can request appointments after completion of 30 days.
* **Admin Dashboard:** Admins can view a list of all submitted TAC applications. Applications can be filtered by category (software, hardware). Admins can view details of each application, approve or reject applications with suitable remarks, and schedule meetings for accepted appointments.
* **Analytics Dashboard**: Admin can view the number of applications by its category. Number of appointments is requested based on the category.

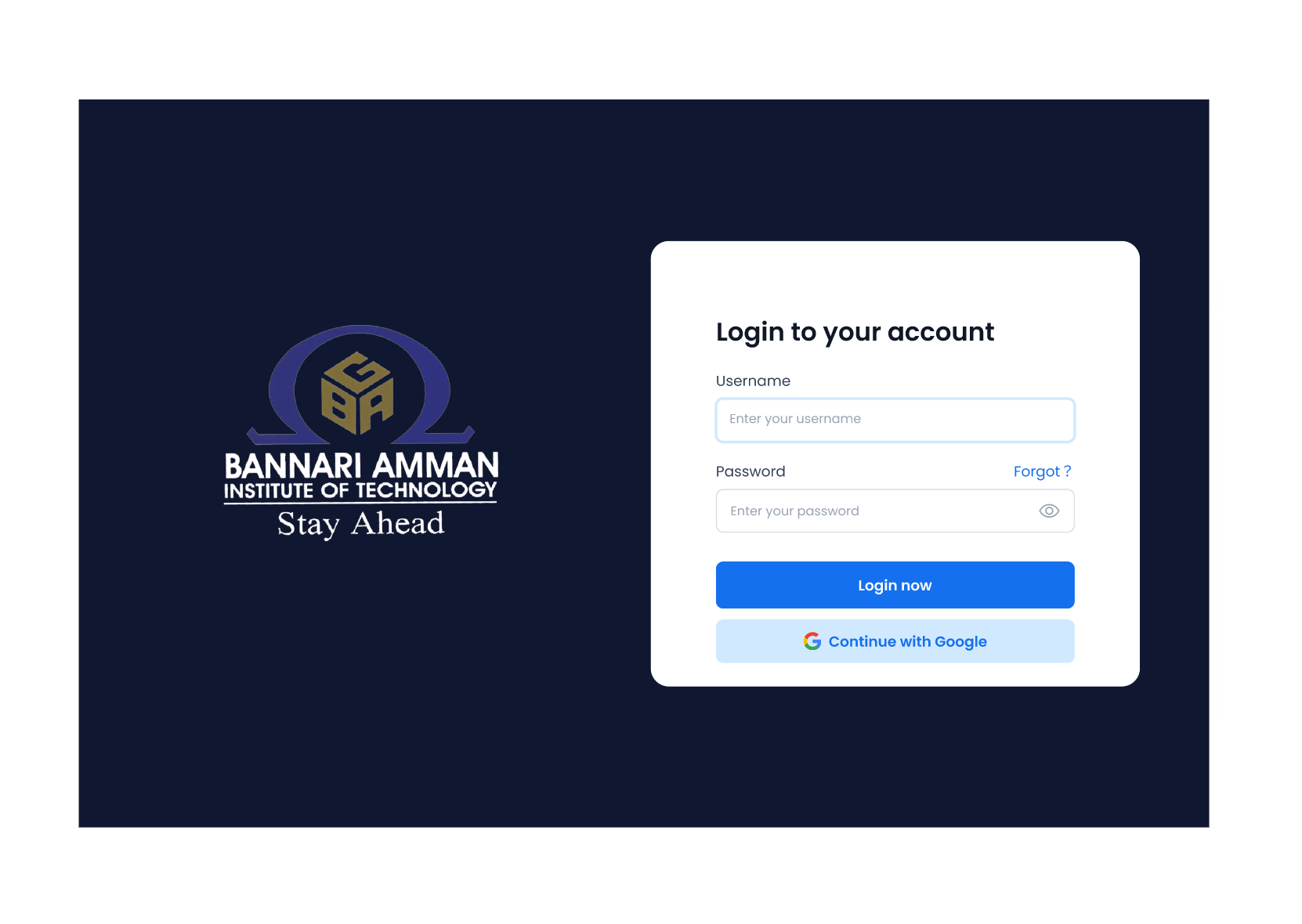
**4. NON-FUNCTIONAL REQUIREMENTS:**

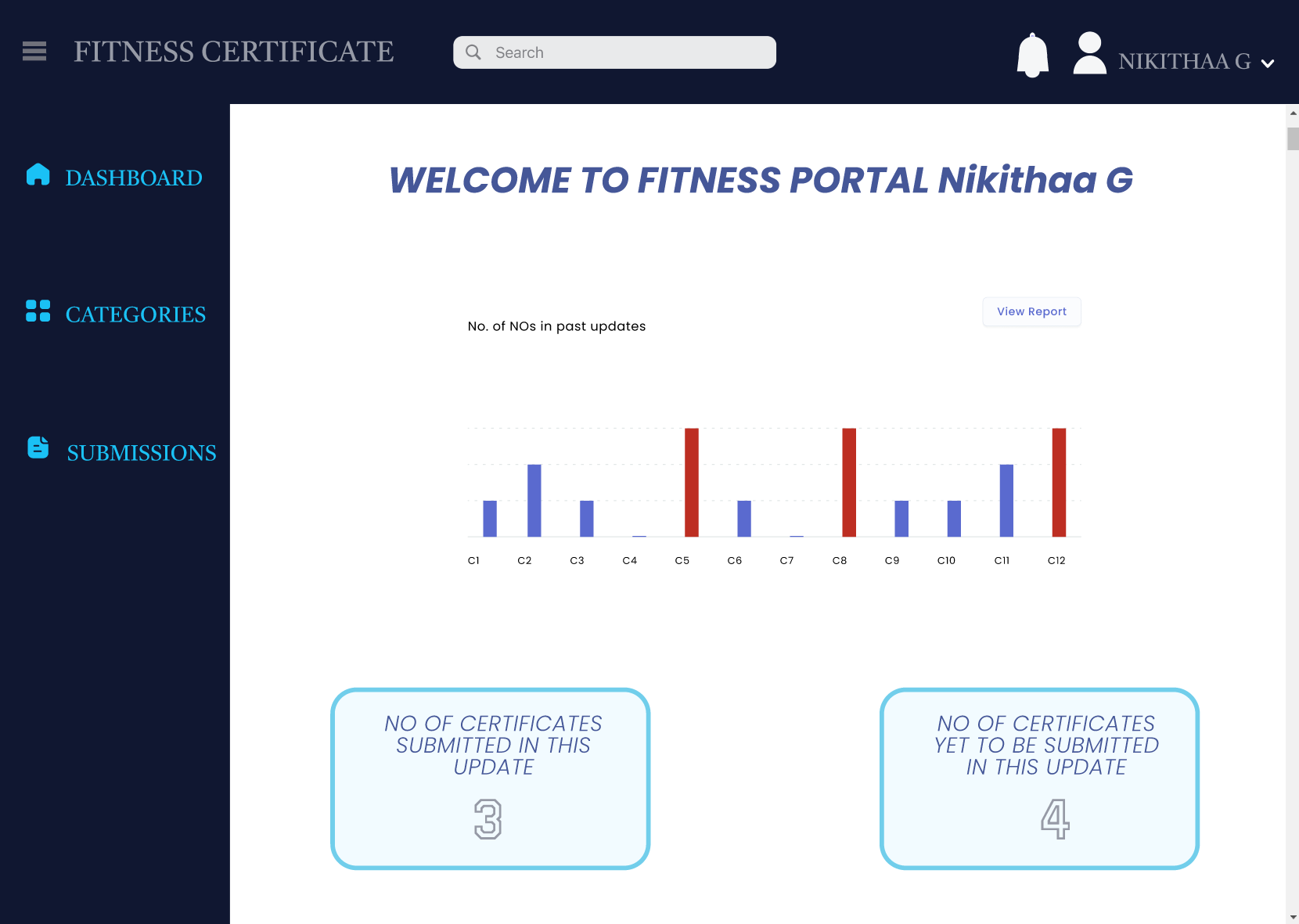
* **Performance**: The system must respond to user actions within 2 seconds. It must handle a concurrent user load of at least 100 users without significant performance degradation.
* **Security**: User data must be encrypted during transmission and storage. Access to sensitive functionalities should be restricted to authorized admin users through secure authentication mechanisms.
* **Usability**: The user interface should be intuitive and user-friendly. Clear and concise error messages should be provided to guide users in case of input errors or system failures.
* **Reliability**: The system should be available 24/7 with minimal downtime. It should have a backup and recovery mechanism in place to prevent data loss in case of system failures or crashes.
* **Scalability**: The system should be designed to accommodate an increasing number of users and data volume over time. It should be scalable to support additional features and functionalities as per future requirements.

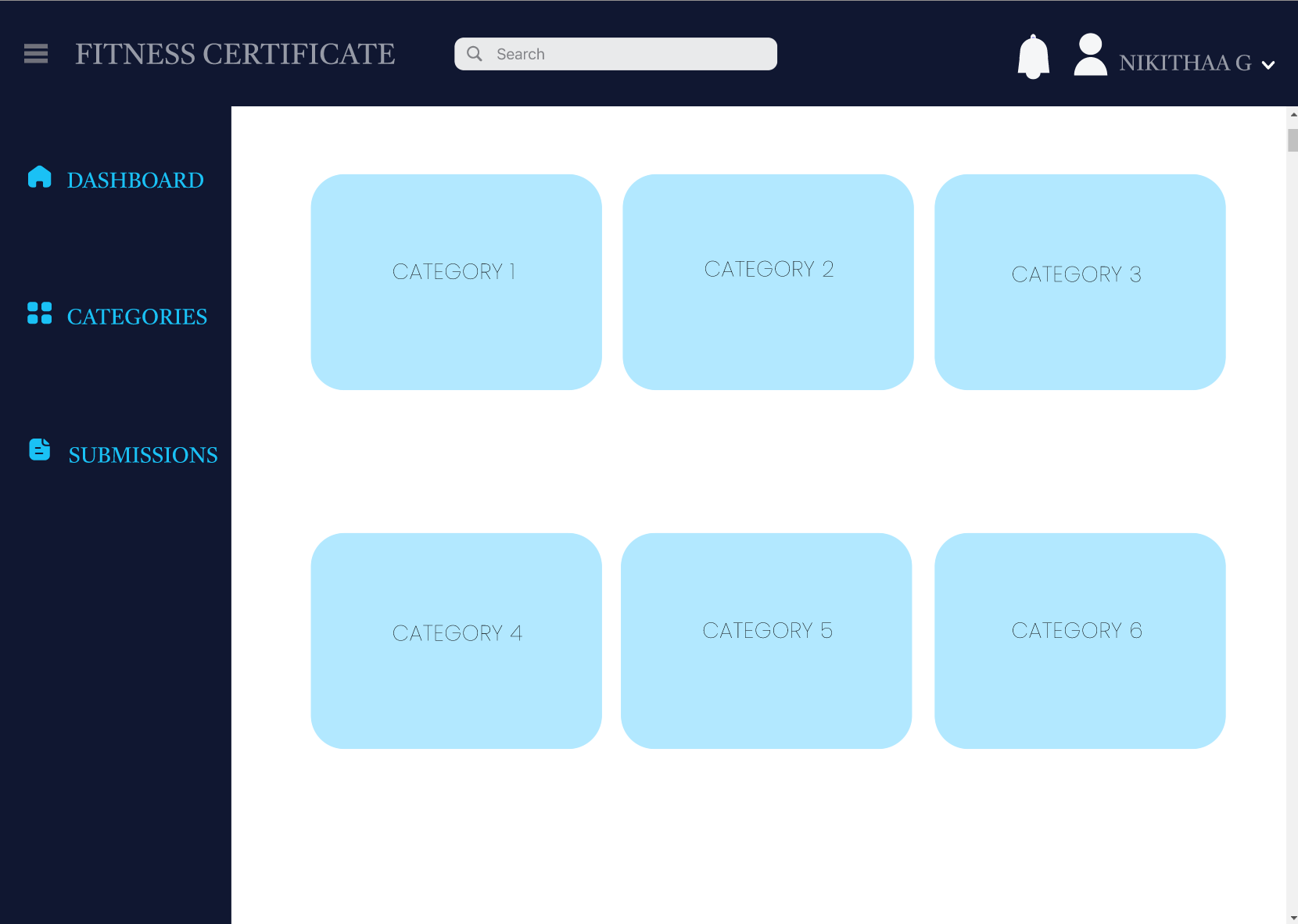
**5. STACK:**

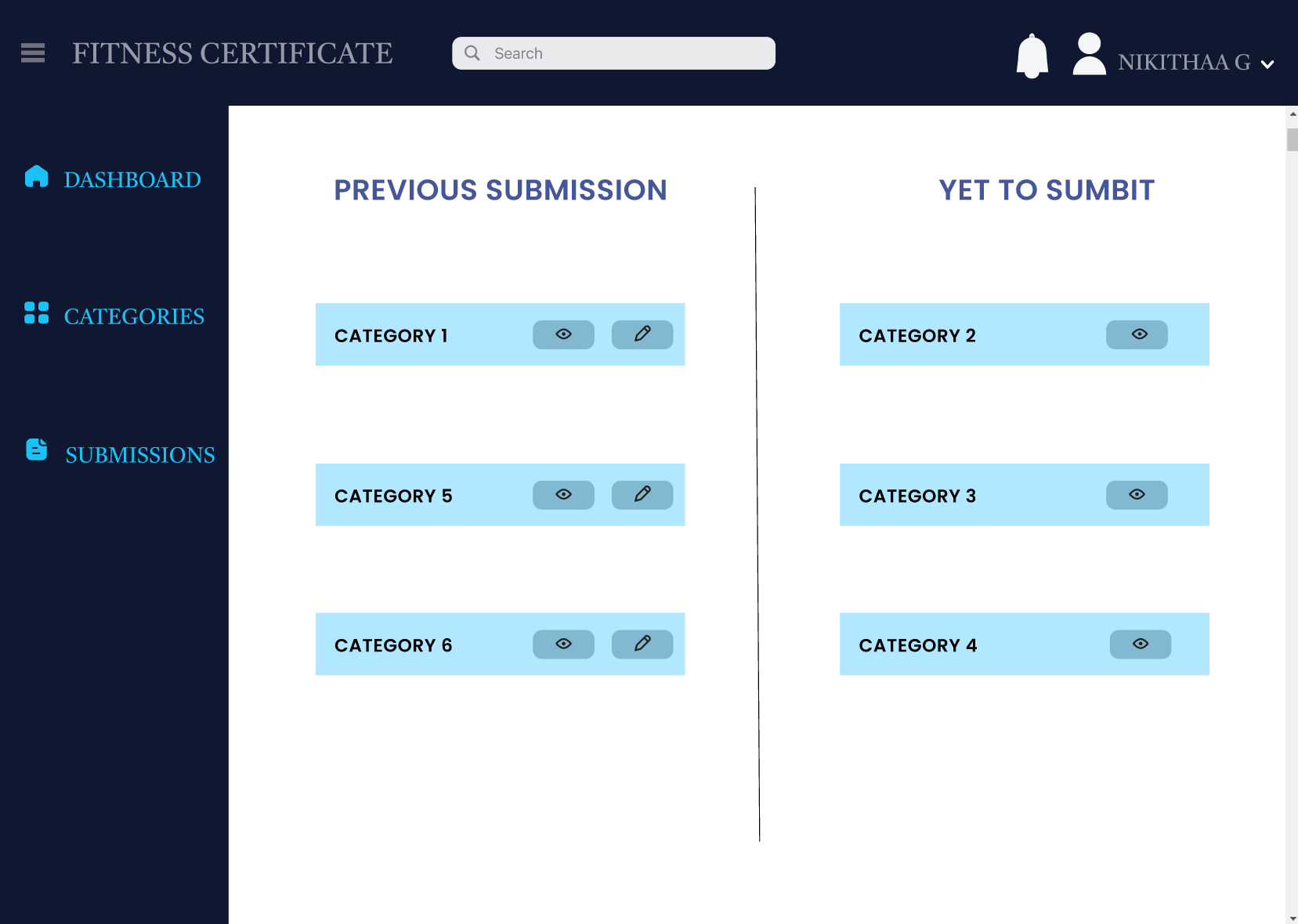
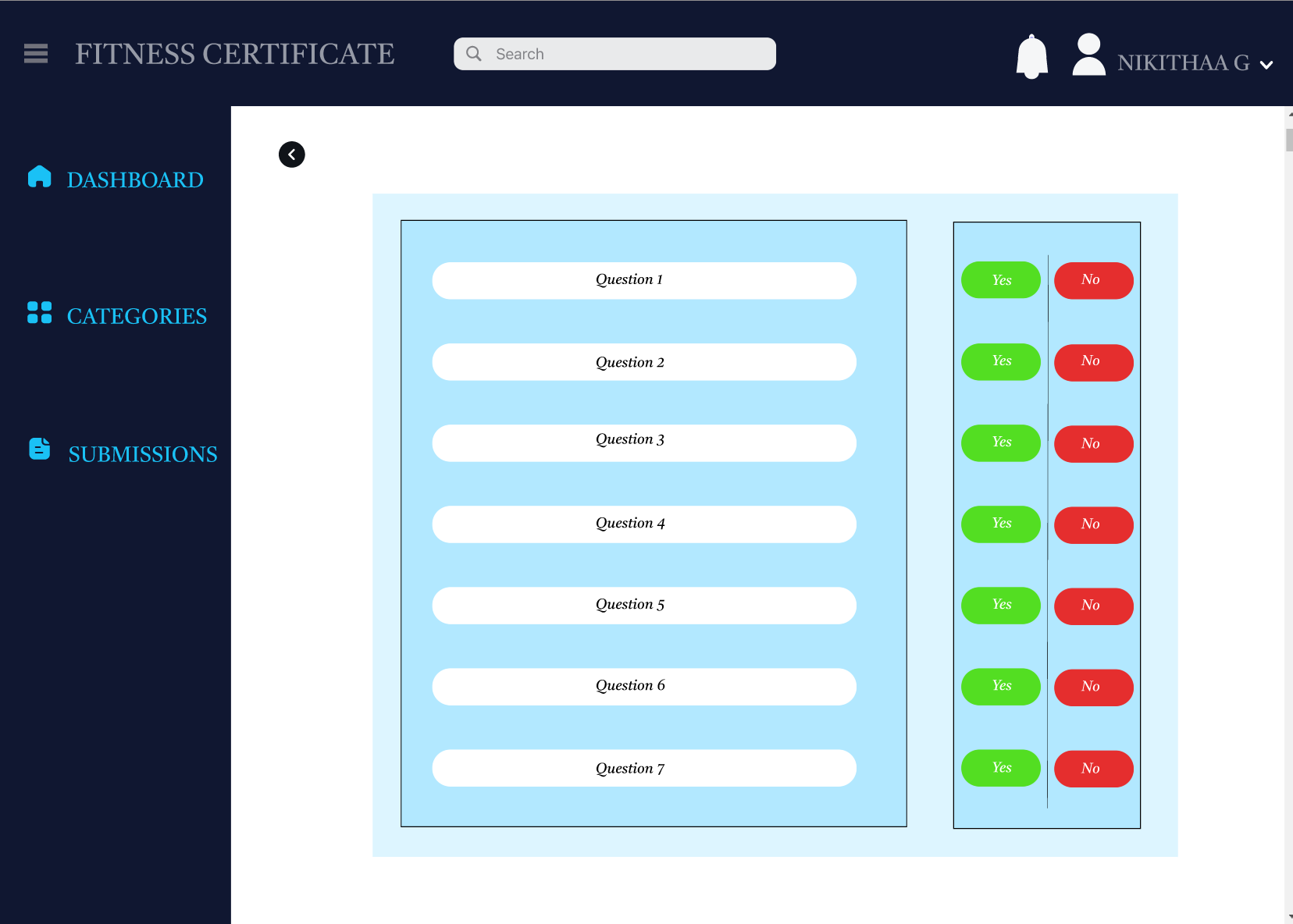
|  |  |
| --- | --- |
| Front End | React (JS Library for building user interfaces) |
| Back End | Node.js with Express.js |
| Database | MongoDB(NOSQL Database) |

**6. PROTOTYPE OF THE PROJECT:**

****

****

****

****