College Club Portal

A Centralized Platform for Student Club Management

Name: Prasanna Somasekar M

Register Number: 43111359

Degree: BE.CSE

Abstract

- The **College Club Portal** is a web-based application designed to streamline the process of discovering and joining student clubs at Sathyabama Institute of Science and Technology.

- It provides students with a **centralized platform** to view available clubs, explore their details (activities, events, descriptions), and register online.

- The system also features a secure **admin panel** for college staff to manage club information and view student registrations efficiently.

- Built using **React** (frontend), **Spring Boot** (backend), and **MongoDB** (database), it offers a modern, user-friendly, and efficient solution compared to traditional methods.

Existing System

- **Current methods** often rely on fragmented information sources:
  - Physical notice boards with posters.
  - Word-of-mouth communication.
  - Individual club social media pages (hard to discover all).
  - Manual registration forms (paper-based or separate Google Forms per club).
- **Problems:**
  - **Difficult Discovery:** Students struggle to find out about *all* available clubs.
  - **Inconsistent Information:** Details might be outdated or vary across platforms.
  - **Inefficient Registration:** Manual forms require significant effort for both students and club coordinators to manage.
  - **Lack of Central Oversight:** Difficult for college administration to get a clear overview of club memberships.

Proposed System

- A **dynamic web application** accessible to all students and administrators.

- **Architecture:**

  - **Frontend:** Built with React for an interactive user interface.

  - **Backend:** A robust Spring Boot application providing a REST API.

  - **Database:** MongoDB stores club details and registration data flexibly.

- **Functionality:**

  - **Homepage:** Displays all clubs with brief descriptions (Club Discovery).

  - **Club Detail Page:** Shows comprehensive information, event history, and images.

  - **Online Registration Form:** Integrated form for easy student sign-up.

  - **Central Database:** All registrations stored in one place (`registrations` collection) with `clubId` linking.

  - **Admin Dashboard:** Secure login (`admin/admin123`) to view and filter all registrations.

Advantages

- **Centralized Information:** One single source for all club-related details.

- **Easy Accessibility:** Students can access the portal anytime, anywhere via a web browser.

- **Simplified Registration:** Quick and easy online form submission.

- **Efficient Management:** Administrators can easily view, filter, and export registration lists per club from MongoDB Compass or the dashboard.

- **Improved Visibility:** Increases awareness of different clubs among the student body.

- **Modern User Experience:** Provides a clean, intuitive interface.

- **Scalability:** Can easily handle adding more clubs or students.

Disadvantages

- **Requires Internet Access:** Both students and administrators need an internet connection.

- **Initial Data Entry:** Populating initial club details and event history requires effort.

- **Dependency on Technology:** Relies on the server and database being operational.

- **(Minor) Learning Curve:** Administrators might need brief training on using the dashboard or MongoDB Compass for filtering/exports.

- **(Currently) Dummy Student Login:** The current student login is a placeholder and doesn't verify against a real student database.

Hardware Requirements

- **Server (for hosting the backend):**
  - Standard Web Server specifications (e.g., Cloud VM or local machine with sufficient RAM - 4GB+, CPU - 2+ cores, Disk Space - 20GB+).

- **Client (for accessing the website):**
  - Any modern computer, laptop, tablet, or smartphone.

- **Database Server:**
  - Machine capable of running MongoDB (can be the same as the web server for smaller loads).

Software Requirements

- **Server:**
  - Operating System (Windows, Linux, macOS)
  - Java Development Kit (JDK) 17 or higher
  - Apache Maven
  - MongoDB Server 5.0 or higher
- **Client:**
  - Modern Web Browser (Edge, Chrome, Firefox, Safari)
- **Development Environment:**
  - IDE like Visual Studio Code, IntelliJ IDEA
  - Node.js and npm
  - Git (for version control)
  - MongoDB Compass (for database management)

Modules

The project can be broken down into these core modules:

1. **Club Browsing Module** (Student Frontend)

2. **Club Details Module** (Student Frontend)

3. **Student Registration Module** (Frontend Form + Backend API)

4. **Authentication Module** (Admin Login - Frontend + Backend)

5. **Admin Dashboard Module** (Admin Frontend + Backend API)

6. **Database Module** (MongoDB Collections: `clubs`, `registrations`, `users`)

Module Description

1. **Club Browsing:** Displays club cards on the homepage, fetched from the /api/clubs endpoint. Allows visual discovery.

2. **Club Details:** Shows detailed info for a single club (fetched via /api/clubs/{id}). Includes description, events, images.

3. **Student Registration:** Provides the form interface (React) and handles form submission to the /api/register backend endpoint, which saves data to MongoDB.

4. **Authentication:** Handles admin login via the /api/auth/login endpoint using Spring Security. Manages user state in the frontend (localStorage).

5. **Admin Dashboard:** A protected route in React that fetches all registrations from /api/admin/registrations and displays them in a table. Includes logout functionality.

6. **Database:** MongoDB stores data in three main collections, managed via Spring Data MongoDB repositories.

Sample Output (Screenshots)

Homepage
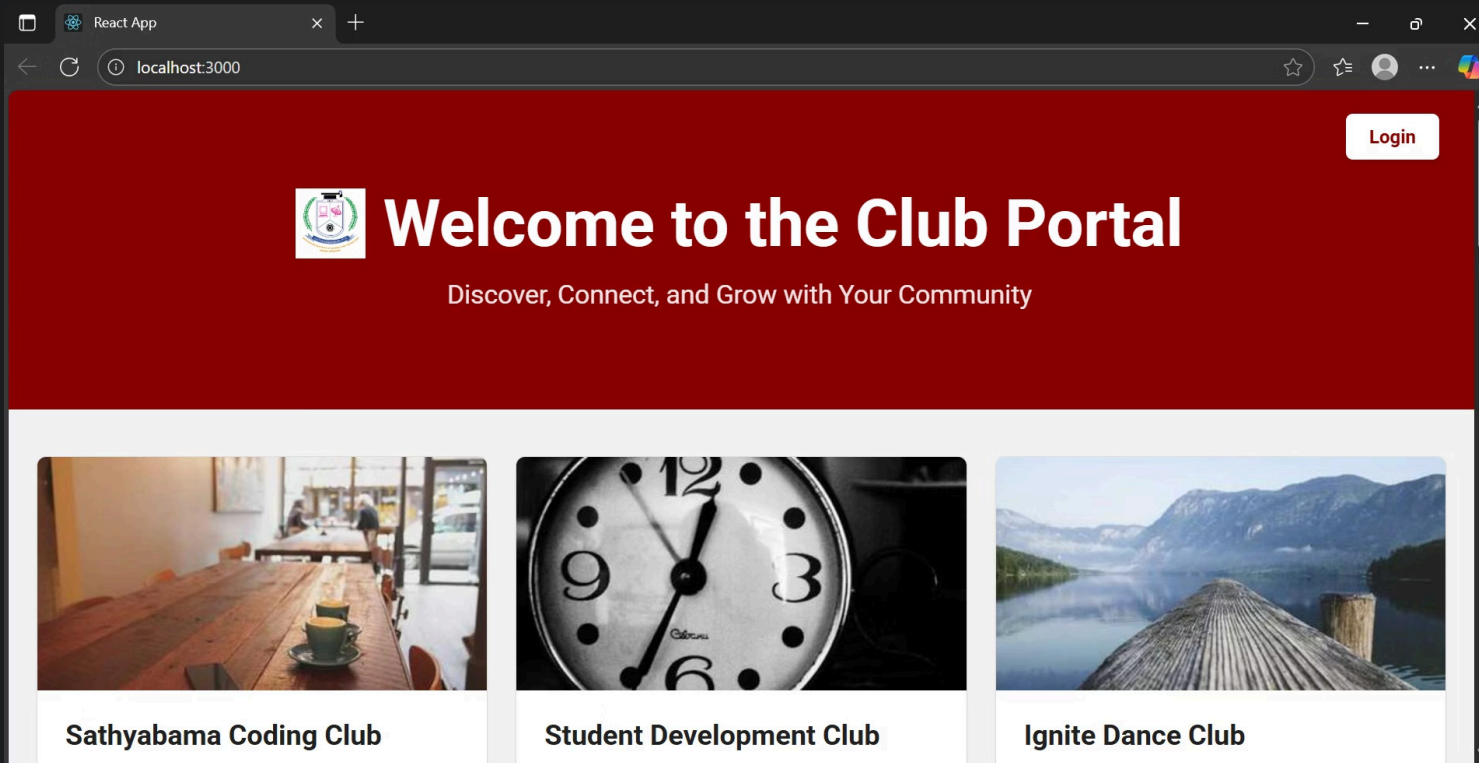
Shows the main portal view with the official Sathyabama-themed header, logo, and login button. Students can immediately see a grid of available clubs with their names and taglines

Club Detail Page

Displays the dedicated page for a specific club (e.g., "Symphony Music Club"). Includes the detailed description, a gallery of past events with images, and the registration form conveniently placed on the side.

Registration Success Confirmation
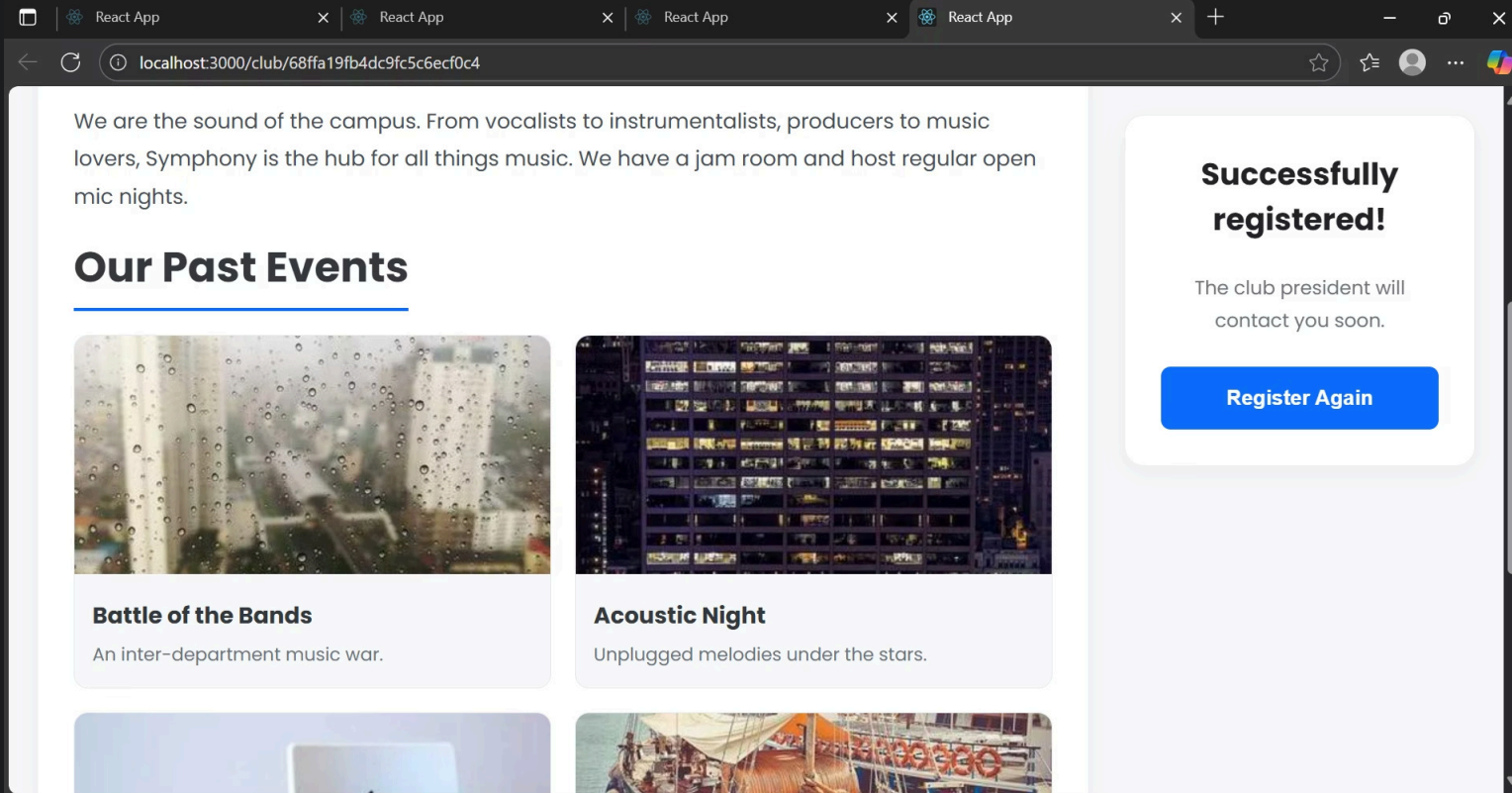
Shows the registration form area after a student successfully submits their details. A clear confirmation message "Successfully registered!" is displayed, informing the student of the next steps.

Professional Login Interface
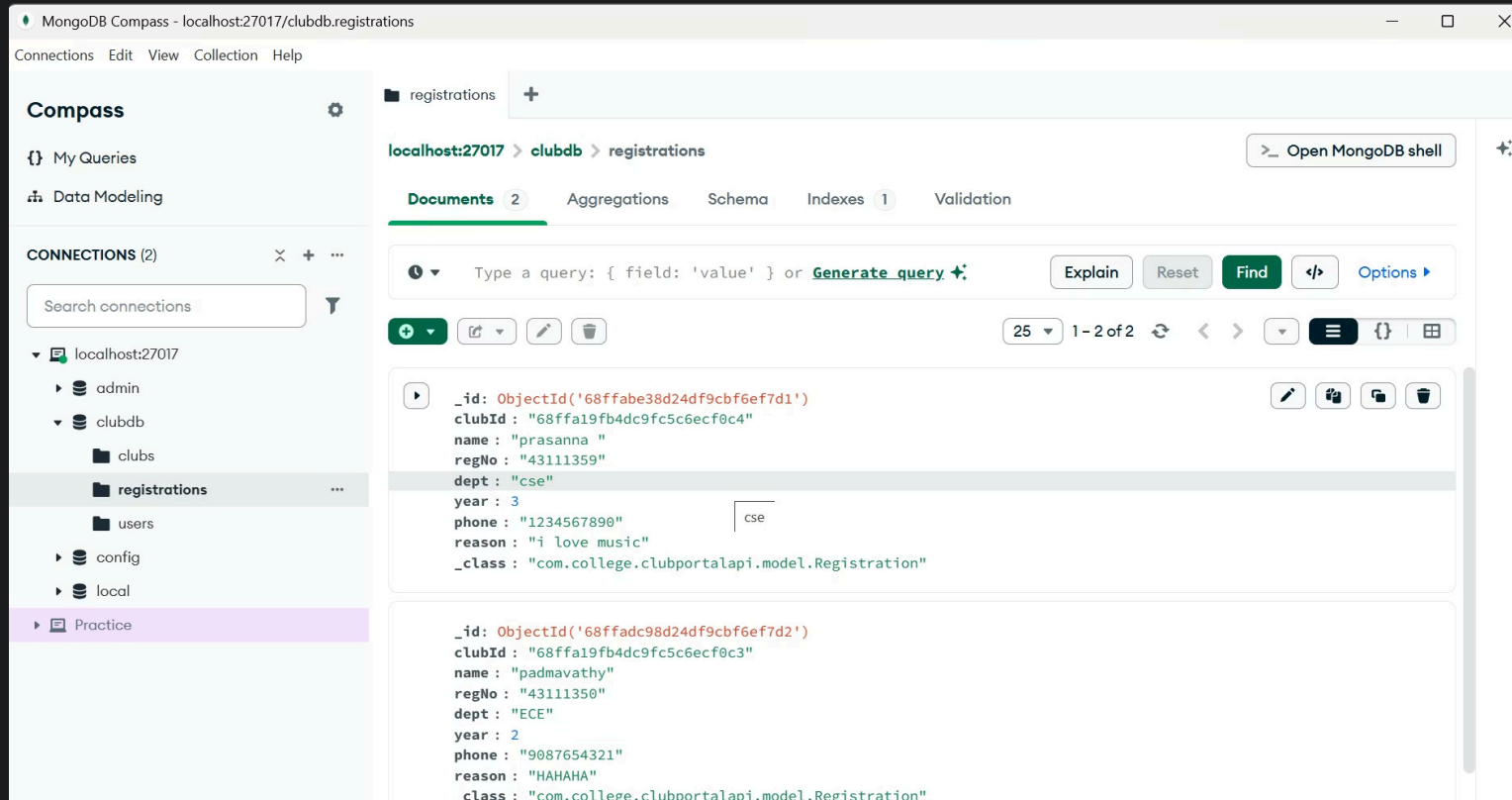

Presents the clean login page with distinct tabs for "Student" (currently a placeholder) and "Admin", allowing different user types to access the system appropriately.

Made with GAMMA

Admin Dashboard - Centralized Data

Displays the secure admin area after successful login. Shows a clear table listing all student registrations received across all clubs, including details like name, register number, department, year, and reason for joining.

Conclusion

- The College Club Portal successfully provides a **centralized, modern solution** for managing student club discovery and registration.

- It **improves efficiency** for students, club coordinators, and administrators.

- The use of React, Spring Boot, and MongoDB provides a **robust and scalable** foundation.

- **Future Enhancements:** Could include real student login integration, club-specific dashboards for presidents, event calendars, or notification systems.