**Synopsis**

**On**

# E-NOTICE BOARD


**BACHELOR OF TECHNOLOGY**
**IN**
**COMPUTER SCIENCE ENGINEERING DEPARTMENT**

**Submitted To:**

**SIGNATURE**
**Mr.Neeraj Raheja**
**Project Coordinator**

**Submitted By:**

**Himanshi** **11222560**
**Mehak Goel** **11222587**

# E-NOTICE BOARD

## CHAPTER 1: INTRODUCTION

### 1.1 Project Overview:

The E-Notice Board is an innovative digital solution designed to revolutionize information dissemination in educational institutions. This web-based application aims to replace traditional physical notice boards with a dynamic, accessible, and efficient platform. By leveraging modern web technologies, the E-Notice Board offers a user-friendly interface for both administrators and students, enhancing communication and information flow within the college community.

### 1.2 Background:

In the digital era, educational institutions are increasingly adopting technology to streamline their processes and improve communication. However, many colleges still rely on outdated methods for sharing important announcements and information. Traditional notice boards, while familiar, have significant limitations in terms of accessibility, timeliness, and organization. The E-Notice Board project emerges from the need to modernize this crucial aspect of college communication.

### 1.3 Problem Statement:

Current notice board systems in many educational institutions face several challenges:

- Limited Accessibility: Physical notice boards are only viewable when students are on campus, restricting access to important information.
- Inefficient Updates: Posting and removing notices manually is time-consuming and can lead to outdated information remaining visible.
- Lack of Organization: It's difficult to categorize and search for specific notices on a traditional board.
- Environmental Concerns: Paper-based notices contribute to waste and are not eco-friendly.
- Ineffective Communication: Important notices may be overlooked among a cluttered physical board.

### 1.4 Proposed Solution

The E-Notice Board addresses these challenges through a comprehensive digital approach:

- User Authentication: Secure login system for administrators and students, ensuring appropriate access control.
- Admin Panel: An intuitive interface for administrators to easily post, edit, and delete notices.

- Categorized Notices: A system to organize notices into categories (e.g., academic, events, general) for easy navigation.
- Search Functionality: Robust search capabilities allowing users to quickly find relevant information.
- Responsive Design: A mobile-friendly interface ensuring access from various devices including smartphones and tablets.
- Date-based Sorting: Automatic organization of notices by date, with options to view past notices.
- Client-side Storage: Utilization of LocalStorage or SessionStorage for efficient data management without the need for a server-side database.
- User-Friendly Interface: An intuitive, easy-to-navigate design for both administrators and students.
- Real-time Updates: Immediate reflection of new notices and updates without page reloads.

By implementing these features, the E-Notice Board aims to significantly improve communication efficiency,enhance information accessibility, and provide a more organized and eco-friendly alternative to traditional notice boards. This solution not only addresses the current problems but also paves the way for future enhancements in college information systems.

# CHAPTER 2:OBJECTIVES

## 2.1 Primary Objectives

1. Develop a User Authentication System:
   a. Implement secure login functionality for administrators and students.
   b. Ensure proper role-based access control.

2. Create an Intuitive Admin Panel:
   a. Design and develop an easy-to-use interface for notice management.
   b. Implement functionalities for posting, editing, and deleting notices.

3. Implement Notice Categorization:
   a. Create a system for organizing notices into different categories.
   b. Develop an interface for easy category management.

4. Design a Responsive User Interface:
   a. Create a mobile-responsive web application.
   b. Ensure easy navigation and access to notices across various devices.

5. Develop Search and Filter Functionality:
   a. Implement a robust search system for finding specific notices.
   b. Create filtering options based on categories and dates.

## 2.2 Secondary Objectives

1. Enhance User Experience:
   a. Implement features like bookmarking important notices.
   b. Develop a notification system for new and urgent notices.

2. Improve Information Management:
   a. Implement date-based sorting and archiving of notices.
   b. Develop a system for managing notice expiration.

3. Provide Data-Driven Insights:
   a. Implement analytics to track notice views and user engagement.
   b. Use data to improve notice visibility and overall system effectiveness.

4. Ensure Data Persistence:
   a. Implement efficient use of LocalStorage or SessionStorage for client-side data management.
   b. Develop robust data handling mechanisms to prevent data loss.

# CHAPTER 3:HARDWARE AND SOFTWARE REQUIREMENTS

## 3.1 Hardware Requirements

- Any device with web browsing capabilities (computers, smartphones, tablets)
- Minimum screen resolution of 320px width (responsive design will adapt to larger screens)

## 3.2 Software Requirements

- Modern web browser (Chrome, Firefox, Safari, Edge - latest versions)
- JavaScript enabled

## 3.3 Development Tools and Technologies

Front-end:

- HTML5 for structure
- CSS3 for styling and animations
- JavaScript (ES6+) for client-side scripting
- Bootstrap 5 for responsive design framework

Back-end:

- LocalStorage or SessionStorage for client-side data persistence

Development Environment:

- Visual Studio Code or any preferred code editor
- Git for version control
- Browser developer tools for testing and debugging

# CHAPTER 4:METHODOLOGY TO BE USED

4.1 Project Phases

1. Planning and Design (1 week):
    - Define project scope and main features
    - Sketch basic user interface designs
2. Development (4 weeks):
    - Week 1-2: Backend development
    - Week 3-4: Frontend development
3. Integration and Testing (2 weeks):
    - Integrate frontend and backend
    - Basic testing and bug fixes
    - Prepare project documentation

4.2 Development Approach:

We will use a linear development process:

- Set up the development environment
- Build the backend API
- Create the frontend user interface
- Integrate frontend and backend
- Implement core features one by one

4.3 Core Features to Implement

- User authentication and authorization
- Admin interface for creating and managing notices
- User interface for viewing and interacting with notices
- Real-time updates and notifications
- Search and filter functionality for notices
- Archive and manage past notices

4.4 Tools and Technologies

- Frontend: React.js, HTML, CSS
- Backend: Node.js with Express
- Database: MongoDB
- Real-Time Updates: Socket.io
- Authentication: JWT (JSON Web Tokens) or OAuth

4.5 Testing

- Manual testing of each feature
- Basic user testing with classmates

4.6 Project Management

- Use a simple to-do list to track tasks
- Regular check-ins with project advisor

## CHAPTER 5 :REFERENCES


- Antunes, F., & Couto, R. (2023). "Smart Campus Transportation: A Review of Intelligent Transport Systems in University Environments." Sustainability, 15(2), 1438.
- React Documentation. https://reactjs.org/docs/getting-started.html
- Node.js Documentation. https://nodejs.org/en/docs/
- MongoDB Manual. https://docs.mongodb.com/manual/
- Socket.io Documentation. https://socket.io/docs/v4/
- JWT Documentation. https://jwt.io/introduction/
- OAuth 2.0 Documentation. https://oauth.net/2/