

CampusCloud

Your student hub for schedules, resources, and campus updates — built simple and fast.

A lightweight, modern web application for managing courses, attendance, students, and academic workflows across Bachelor of Computer Applications (BCA) and Master of Computer Applications (MCA) programs.

⌚ Project Overview

CampusCloud is an educational management system designed to streamline campus operations by providing intuitive interfaces for:

- **Course Management:** Add, edit, and delete courses for different academic terms.
 - **Student Management:** Manage student records with role-based access control.
 - **Attendance Tracking:** Mark, review, and manage student attendance with date-wise records.
 - **Task Approvals:** Admin-driven workflow for course/student additions, edits, and deletions.
 - **Role-Based Access:** Admin, Moderator, Instructor, and User roles with granular permissions.
-

❖ Key Features

1. Authentication & Authorization

- Secure login system with role-based access control (RBAC).
- Supported roles: Admin, Moderator, Instructor, User.
- Session management and automatic logout.
- Password-protected course deletion with typed confirmation.

2. Course Management

- **Add Courses:** Submit course requests for admin approval.
- **Edit Courses:** Request modifications to existing courses.
- **Delete Courses:** Immediate deletion with typed confirmation phrase ("yes i want to delete it").
- Support for multiple academic terms: BCA I–VI, MCA I–VI.
- Store course details: code, subject, type, credits, internal/external marks, instructor.

3. Student Management

- Add, edit, and delete student records.
- Track student data: exam roll number, name, contact, parent contact, etc.
- Requests require admin approval before implementation.
- Cleanup of non-pending tasks to prevent duplicates.

4. Attendance System

- **Per-Course Attendance:** Mark attendance for BCA and MCA courses (student-wise, date-wise).
- **Overall Attendance:** View and manage attendance across all courses, grouped by date.

- **Moderator View:** Read-only summary of attendance records across the system.
- **Date-Wise Summaries:** See attendance history per course with edit/view options.
- **Persistent Storage:** Attendance records are saved with unique constraint per date/student/course.
- **Permission-Based Editing:** Only instructors and admins can submit/edit attendance.

5. Task Approval Workflow

- Centralized dashboard for admins to review pending requests.
- Approve, reject, or manage course and student tasks.
- Task history with timestamps and submitter information.
- Snapshot storage for audit trails.

6. User Interface & Experience

- **Responsive Design:** Works seamlessly on desktop, tablet, and mobile devices.
- **Dark Mode Support:** Built-in light/dark theme toggle with `theme.js` (localStorage persistence).
- **Modern Styling:** Glass-morphism effects, smooth transitions, and accessible focus rings.
- **Theme-Aware CSS Variables:** Consistent color palette across all pages.
- **Accessible UI:** Proper focus management, ARIA labels, and semantic HTML.

7. Dashboard

- Personalized dashboard for admin/moderator showing pending tasks.
- Quick stats on courses, students, and attendance.
- Task cards with status badges (pending, approved, rejected).
- Profile management for admins.

🛠️ Technology Stack

Layer	Technology
Backend	PHP (procedural & OOP patterns)
Database	MySQL (mysqli)
Frontend	HTML5, CSS3, JavaScript (vanilla)
Server	Apache (XAMPP)
Theme System	CSS Variables + theme.js

📁 Project Structure

```
CampusCloud/
  └── auth/
      ├── auth.php          # Login page
      ├── landing.php       # Public landing page
      ├── dashboard.php     # Admin/moderator dashboard
      └── delete_course.php # Immediate course deletion with confirmation
```

```

├── add_course.php          # Add course request
├── update_course.php       # Edit course request
├── courses_bca.php        # BCA courses management
├── courses_mca.php        # MCA courses management
├── students_bca.php       # BCA student management
├── students_mca.php       # MCA student management
├── attendance_bca.php     # BCA attendance marking
├── attendance_mca.php     # MCA attendance marking
├── overall_attendance.php # Overall attendance (all courses, datewise)
├── overall_attendance_moderator.php # Moderator view (read-only)
├── approve_tasks.php      # Admin task approval interface
├── tasks_history.php       # Task history & audit logs
├── require_role.php        # Role enforcement helper
└── logout.php              # Session logout

logs/
└── delete_course_debug.log # Debug logs for delete operations

styles/
├── main.css                # Core theme & variables (light/dark)
├── landing.css              # Landing page styles
├── login.css                # Login form styles
├── dashboard.css            # Dashboard layout
├── courses.css              # Course cards & tables
├── students.css              # Student form & tables
├── add_course.css           # Add course form styles
├── attendance.css           # Attendance modal & controls
├── tasks_history.css        # Tasks history layout
├── internal.css             # Internal marks styles
├── faculty.css              # Faculty/instructor styles
└── theme.js                 # Theme toggle & persistence

db/
└── connection.php           # MySQL database connection

api/
└── setup_tables.php         # Database schema initialization

README.md                  # This file

```

🚀 Getting Started

Prerequisites

- **XAMPP** (or similar Apache + PHP + MySQL stack)
- **PHP 7.4+**
- **MySQL 5.7+**
- **Modern web browser** (Chrome, Firefox, Safari, Edge)

Installation

1. **Extract the project** to your XAMPP `htdocs` folder:

```
C:\xampp\htdocs\CampusCloud
```

2. Start XAMPP services:

- Start Apache
- Start MySQL

3. Initialize the database:

- Open <http://localhost/phpmyadmin>
- Create a new database named `campuscloud`
- Import or run the schema (or navigate to the setup endpoint)

4. Run database setup (creates tables):

- Navigate to: http://localhost/CampusCloud/api/setup_tables.php
- Expected response: `{"status": "OK", "errors": [], "success": [...]}`

5. Access the application:

- Landing page: <http://localhost/CampusCloud/auth/landing.php>
 - Login: <http://localhost/CampusCloud/auth/auth.php>
-

User Roles & Permissions

Role	Capabilities
Admin	Full system access; approve/reject tasks; manage all courses/students; view all attendance.
Moderator	Read-only dashboard; view pending tasks; no direct editing permissions.
Instructor	Mark attendance for assigned courses; view course details; limited student visibility.
User	Limited access; view personal course info; submit course/student requests for approval.

Key Workflows

Adding a Course

1. User navigates to **Add Course** page.
2. Fills in course details (code, subject, credits, etc.).
3. Submits → creates a task record (pending approval).
4. Admin reviews on **Approve Tasks** page.
5. Admin approves → course is added to the respective table (BCA/MCA).

Marking Attendance

1. Instructor opens **BCA/MCA Attendance** page.
2. Selects a date via date picker.
3. Loads existing records or starts fresh.
4. Marks students as Present/Absent/Leave using radio buttons.
5. Saves → records stored in `attendance_records` table with unique constraint per (term, course, student, date).

6. Can view/edit date-wise summaries in the same page.

Deleting a Course

1. Instructor clicks **Delete** on a course.
2. Redirected to confirmation page showing course details.
3. Must type "yes i want to delete it" exactly (case-insensitive).
4. Clicks **Delete Permanently**.
5. Course is immediately deleted from the course table (no approval needed).
6. Deletion logged to [auth/logs/delete_course_debug.log](#).

Viewing Overall Attendance

1. User opens **Overall Attendance** page.
 2. Sees all courses, grouped by semester/term.
 3. Selects a date to filter or load specific date's records.
 4. Admin/Instructor can edit; others see read-only summary.
 5. Moderator has dedicated read-only view.
-

Database Schema

Core Tables

Table	Purpose
users	User credentials, roles, timestamps
bca_courses, mca_courses	Course records per term
bca_student_i to bca_student_vi, mca_student_i to mca_student_vi	Student records per term
attendance_records	Attendance entries (term, course, student, date, status)
tasks	Course-level task requests (add, edit, delete)
task_details	Course details snapshot for task audit
student_tasks	Student-level task requests
student_task_details	Student details snapshot for task audit

Key Constraints

- **Attendance UNIQUE constraint:** ([term, course_table, course_id, student_table, student_id, att_date](#)) ensures one record per date/course/student.
 - **Foreign keys:** [submitted_by](#) references [users.user_id](#) (ON DELETE SET NULL).
-

Theme System

The application uses **CSS custom properties (variables)** for theming:

```
:root {  
    --bg: #f8fafc;          /* Background */  
    --surface: #ffffff;      /* Cards/surfaces */  
    --text: #0f172a;         /* Text color */  
    --primary: #4f46e5;      /* Primary accent */  
    --danger: #ef4444;        /* Danger/delete color */  
    --muted: #64748b;        /* Muted text */  
    /* ... more variables */  
}  
  
.dark {  
    --bg: #0f172a;          /* Dark background */  
    --surface: #1e293b;        /* Dark surface */  
    /* ... dark mode overrides */  
}
```

Theme Toggle: Click the  button to switch themes. Preference is saved to browser's `localStorage` as `cc_theme`.

🔧 Configuration

Database Connection

Edit `db/connection.php`:

```
$host = 'localhost';  
$user = 'root';  
$password = 'your_password';  
$db_name = 'campuscloud';
```

Debug Mode

Enable debug output for delete operations:

```
http://localhost/CampusCloud/auth/delete_course.php?  
table=bca_courses&id=123&debug=1
```

Check logs: `auth/logs/delete_course_debug.log`

✍ Testing

Manual Test Checklist

- Login with admin/moderator credentials
 - Add a course and verify approval workflow
 - Edit a course and check task status
 - Delete a course with confirmation
 - Mark attendance for a date
 - View attendance history and edit previous records
 - Switch between light and dark themes
 - Test on mobile/tablet (responsive design)
 - Verify role-based access restrictions
 - Check attendance unique constraint (try duplicate entry)
-

API Endpoints

Setup & Initialization

- **POST** `/api/setup_tables.php` → Creates/verifies all tables; returns JSON status.

Attendance (AJAX Endpoints)

- **POST** `/auth/attendance_bca.php?action=load_attendance` → Load records for a date.
 - **POST** `/auth/attendance_bca.php?action=save_attendance` → Save/update attendance records.
 - **POST** `/auth/overall_attendance.php?action=load_attendance` → Load overall attendance.
 - **POST** `/auth/overall_attendance.php?action=update_attendance` → Update row in overall view.
-

Troubleshooting

Issue: "Database connection failed"

- Verify MySQL is running in XAMPP.
- Check credentials in `db/connection.php`.
- Ensure database `campuscloud` exists.

Issue: "Course not found" on delete

- Confirm course ID is valid in the URL.
- Verify course exists in the respective course table (`bca_courses/mca_courses`).

Issue: Attendance not saving

- Check browser console for JavaScript errors.
- Verify user has instructor/admin role.
- Ensure date parameter is sent in POST request.
- Check `auth/logs/delete_course_debug.log` for DB errors.

Issue: Theme not persisting

- Clear browser cache and localStorage.
- Verify `theme.js` is loaded (check network tab in DevTools).

- Ensure `main.css` is linked correctly.
-

📞 Support & Contact

For questions or issues, please refer to the project documentation or contact the development team.

📄 License

This project is proprietary and confidential. Unauthorized copying or modification is prohibited.

🎓 Credits

Developed as an educational management system for academic institutions.

Last Updated: December 7, 2025

Version: 1.0.0