<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>School Management System</title>

<script src="https://cdn.tailwindcss.com"></script>

<style>

@import url('https://fonts.googleapis.com/css2?family=Inter:wght@300;400;500;600;700&display=swap');

body { font-family: 'Inter', sans-serif; }

.tab-content { display: none; }

.tab-content.active { display: block; }

.sidebar-item:hover { background-color: rgba(59, 130, 246, 0.1); }

.sidebar-item.active { background-color: rgba(59, 130, 246, 0.2); border-right: 3px solid #3b82f6; }

</style>

</head>

<body class="bg-gray-50 min-h-screen">

<!-- Header -->

<header class="bg-blue-600 text-white shadow-lg">

<div class="container mx-auto px-6 py-4">

<div class="flex items-center justify-between">

<div class="flex items-center space-x-3">

<div class="w-10 h-10 bg-white rounded-lg flex items-center justify-center">

<span class="text-blue-600 font-bold text-xl">📚</span>

</div>

<div>

<h1 class="text-2xl font-bold">EduManage Pro</h1>

<p class="text-blue-100 text-sm">School Management System</p>

</div>

</div>

<div class="flex items-center space-x-4">

<span class="text-blue-100">Welcome, Admin</span>

<div class="w-8 h-8 bg-blue-500 rounded-full flex items-center justify-center">

<span class="text-sm font-semibold">A</span>

</div>

</div>

</div>

</div>

</header>

<div class="flex">

<!-- Sidebar -->

<aside class="w-64 bg-white shadow-lg min-h-screen">

<nav class="mt-6">

<div class="px-6 py-3">

<h3 class="text-gray-500 text-xs font-semibold uppercase tracking-wider">Main Menu</h3>

</div>

<div class="space-y-1">

<button onclick="showTab('dashboard')" class="sidebar-item active w-full text-left px-6 py-3 text-gray-700 hover:text-blue-600 transition-colors duration-200 flex items-center space-x-3">

<span>📊</span>

<span>Dashboard</span>

</button>

<button onclick="showTab('students')" class="sidebar-item w-full text-left px-6 py-3 text-gray-700 hover:text-blue-600 transition-colors duration-200 flex items-center space-x-3">

<span>👥</span>

<span>Students</span>

</button>

<button onclick="showTab('teachers')" class="sidebar-item w-full text-left px-6 py-3 text-gray-700 hover:text-blue-600 transition-colors duration-200 flex items-center space-x-3">

<span>👨‍🏫</span>

<span>Teachers</span>

</button>

<button onclick="showTab('courses')" class="sidebar-item w-full text-left px-6 py-3 text-gray-700 hover:text-blue-600 transition-colors duration-200 flex items-center space-x-3">

<span>📖</span>

<span>Courses</span>

</button>

<button onclick="showTab('grades')" class="sidebar-item w-full text-left px-6 py-3 text-gray-700 hover:text-blue-600 transition-colors duration-200 flex items-center space-x-3">

<span>📝</span>

<span>Grades</span>

</button>

<button onclick="showTab('attendance')" class="sidebar-item w-full text-left px-6 py-3 text-gray-700 hover:text-blue-600 transition-colors duration-200 flex items-center space-x-3">

<span>📅</span>

<span>Attendance</span>

</button>

<button onclick="showTab('reports')" class="sidebar-item w-full text-left px-6 py-3 text-gray-700 hover:text-blue-600 transition-colors duration-200 flex items-center space-x-3">

<span>📊</span>

<span>Reports</span>

</button>

</div>

</nav>

</aside>

<!-- Main Content -->

<main class="flex-1 p-6">

<!-- Dashboard Tab -->

<div id="dashboard" class="tab-content active">

<div class="mb-6">

<h2 class="text-3xl font-bold text-gray-800">Dashboard</h2>

<p class="text-gray-600">Overview of your school management system</p>

</div>

<!-- Stats Cards -->

<div class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-4 gap-6 mb-8">

<div class="bg-white rounded-lg shadow-md p-6 border-l-4 border-blue-500">

<div class="flex items-center justify-between">

<div>

<p class="text-gray-500 text-sm font-medium">Total Students</p>

<p class="text-3xl font-bold text-gray-800" id="totalStudents">1,247</p>

</div>

<div class="w-12 h-12 bg-blue-100 rounded-lg flex items-center justify-center">

<span class="text-blue-600 text-xl">👥</span>

</div>

</div>

</div>

<div class="bg-white rounded-lg shadow-md p-6 border-l-4 border-green-500">

<div class="flex items-center justify-between">

<div>

<p class="text-gray-500 text-sm font-medium">Total Teachers</p>

<p class="text-3xl font-bold text-gray-800" id="totalTeachers">89</p>

</div>

<div class="w-12 h-12 bg-green-100 rounded-lg flex items-center justify-center">

<span class="text-green-600 text-xl">👨‍🏫</span>

</div>

</div>

</div>

<div class="bg-white rounded-lg shadow-md p-6 border-l-4 border-purple-500">

<div class="flex items-center justify-between">

<div>

<p class="text-gray-500 text-sm font-medium">Active Courses</p>

<p class="text-3xl font-bold text-gray-800" id="totalCourses">24</p>

</div>

<div class="w-12 h-12 bg-purple-100 rounded-lg flex items-center justify-center">

<span class="text-purple-600 text-xl">📖</span>

</div>

</div>

</div>

<div class="bg-white rounded-lg shadow-md p-6 border-l-4 border-orange-500">

<div class="flex items-center justify-between">

<div>

<p class="text-gray-500 text-sm font-medium">Attendance Rate</p>

<p class="text-3xl font-bold text-gray-800">94.2%</p>

</div>

<div class="w-12 h-12 bg-orange-100 rounded-lg flex items-center justify-center">

<span class="text-orange-600 text-xl">📅</span>

</div>

</div>

</div>

</div>

<!-- Recent Activity -->

<div class="bg-white rounded-lg shadow-md p-6">

<h3 class="text-xl font-semibold text-gray-800 mb-4">Recent Activity</h3>

<div class="space-y-4">

<div class="flex items-center space-x-4 p-3 bg-gray-50 rounded-lg">

<div class="w-10 h-10 bg-blue-100 rounded-full flex items-center justify-center">

<span class="text-blue-600">👤</span>

</div>

<div class="flex-1">

<p class="font-medium text-gray-800">New student enrolled: Sarah Johnson</p>

<p class="text-sm text-gray-500">2 hours ago</p>

</div>

</div>

<div class="flex items-center space-x-4 p-3 bg-gray-50 rounded-lg">

<div class="w-10 h-10 bg-green-100 rounded-full flex items-center justify-center">

<span class="text-green-600">📝</span>

</div>

<div class="flex-1">

<p class="font-medium text-gray-800">Grades updated for Mathematics Class</p>

<p class="text-sm text-gray-500">4 hours ago</p>

</div>

</div>

<div class="flex items-center space-x-4 p-3 bg-gray-50 rounded-lg">

<div class="w-10 h-10 bg-purple-100 rounded-full flex items-center justify-center">

<span class="text-purple-600">👨‍🏫</span>

</div>

<div class="flex-1">

<p class="font-medium text-gray-800">New teacher added: Dr. Michael Brown</p>

<p class="text-sm text-gray-500">1 day ago</p>

</div>

</div>

</div>

</div>

</div>

<!-- Students Tab -->

<div id="students" class="tab-content">

<div class="mb-6 flex justify-between items-center">

<div>

<h2 class="text-3xl font-bold text-gray-800">Students</h2>

<p class="text-gray-600">Manage student records and information</p>

</div>

<button onclick="showAddStudentModal()" class="bg-blue-600 hover:bg-blue-700 text-white px-6 py-2 rounded-lg font-medium transition-colors duration-200">

+ Add Student

</button>

</div>

<!-- Search and Filter -->

<div class="bg-white rounded-lg shadow-md p-6 mb-6">

<div class="flex flex-col md:flex-row gap-4">

<div class="flex-1">

<input type="text" id="studentSearch" placeholder="Search students..." class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent" onkeyup="filterStudents()">

</div>

<select id="gradeFilter" class="px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent" onchange="filterStudents()">

<option value="">All Grades</option>

<option value="9">Grade 9</option>

<option value="10">Grade 10</option>

<option value="11">Grade 11</option>

<option value="12">Grade 12</option>

</select>

</div>

</div>

<!-- Students Table -->

<div class="bg-white rounded-lg shadow-md overflow-hidden">

<table class="w-full">

<thead class="bg-gray-50">

<tr>

<th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Student</th>

<th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">ID</th>

<th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Grade</th>

<th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Email</th>

<th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Status</th>

<th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Actions</th>

</tr>

</thead>

<tbody id="studentsTableBody" class="bg-white divide-y divide-gray-200">

<!-- Students will be populated here -->

</tbody>

</table>

</div>

</div>

<!-- Teachers Tab -->

<div id="teachers" class="tab-content">

<div class="mb-6 flex justify-between items-center">

<div>

<h2 class="text-3xl font-bold text-gray-800">Teachers</h2>

<p class="text-gray-600">Manage teaching staff and their assignments</p>

</div>

<button onclick="showAddTeacherModal()" class="bg-green-600 hover:bg-green-700 text-white px-6 py-2 rounded-lg font-medium transition-colors duration-200">

+ Add Teacher

</button>

</div>

<!-- Teachers Grid -->

<div id="teachersGrid" class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-6">

<!-- Teachers will be populated here -->

</div>

</div>

<!-- Courses Tab -->

<div id="courses" class="tab-content">

<div class="mb-6 flex justify-between items-center">

<div>

<h2 class="text-3xl font-bold text-gray-800">Courses</h2>

<p class="text-gray-600">Manage course curriculum and schedules</p>

</div>

<button onclick="showAddCourseModal()" class="bg-purple-600 hover:bg-purple-700 text-white px-6 py-2 rounded-lg font-medium transition-colors duration-200">

+ Add Course

</button>

</div>

<!-- Courses Grid -->

<div id="coursesGrid" class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-6">

<!-- Courses will be populated here -->

</div>

</div>

<!-- Grades Tab -->

<div id="grades" class="tab-content">

<div class="mb-6">

<h2 class="text-3xl font-bold text-gray-800">Grades Management</h2>

<p class="text-gray-600">Record and manage student grades</p>

</div>

<!-- Grade Entry Form -->

<div class="bg-white rounded-lg shadow-md p-6 mb-6">

<h3 class="text-xl font-semibold text-gray-800 mb-4">Enter Grades</h3>

<div class="grid grid-cols-1 md:grid-cols-3 gap-4">

<select id="gradeStudent" class="px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

<option value="">Select Student</option>

</select>

<select id="gradeCourse" class="px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

<option value="">Select Course</option>

</select>

<div class="flex gap-2">

<input type="number" id="gradeValue" placeholder="Grade (0-100)" min="0" max="100" class="flex-1 px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

<button onclick="addGrade()" class="bg-blue-600 hover:bg-blue-700 text-white px-6 py-2 rounded-lg font-medium transition-colors duration-200">

Add Grade

</button>

</div>

</div>

</div>

<!-- Grades Table -->

<div class="bg-white rounded-lg shadow-md overflow-hidden">

<table class="w-full">

<thead class="bg-gray-50">

<tr>

<th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Student</th>

<th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Course</th>

<th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Grade</th>

<th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Letter Grade</th>

<th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Date</th>

<th class="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase tracking-wider">Actions</th>

</tr>

</thead>

<tbody id="gradesTableBody" class="bg-white divide-y divide-gray-200">

<!-- Grades will be populated here -->

</tbody>

</table>

</div>

</div>

<!-- Attendance Tab -->

<div id="attendance" class="tab-content">

<div class="mb-6">

<h2 class="text-3xl font-bold text-gray-800">Attendance Management</h2>

<p class="text-gray-600">Track and manage student attendance</p>

</div>

<!-- Attendance Controls -->

<div class="bg-white rounded-lg shadow-md p-6 mb-6">

<div class="flex flex-col md:flex-row gap-4 items-end">

<div class="flex-1">

<label class="block text-sm font-medium text-gray-700 mb-2">Select Date</label>

<input type="date" id="attendanceDate" class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

</div>

<div class="flex-1">

<label class="block text-sm font-medium text-gray-700 mb-2">Select Course</label>

<select id="attendanceCourse" class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

<option value="">Select Course</option>

</select>

</div>

<button onclick="loadAttendance()" class="bg-blue-600 hover:bg-blue-700 text-white px-6 py-2 rounded-lg font-medium transition-colors duration-200">

Load Attendance

</button>

</div>

</div>

<!-- Attendance List -->

<div id="attendanceList" class="bg-white rounded-lg shadow-md p-6">

<h3 class="text-xl font-semibold text-gray-800 mb-4">Student Attendance</h3>

<div id="attendanceStudents" class="space-y-3">

<p class="text-gray-500">Select a date and course to load attendance</p>

</div>

<div class="mt-6 flex gap-4">

<button onclick="markAllPresent()" class="bg-green-600 hover:bg-green-700 text-white px-4 py-2 rounded-lg font-medium transition-colors duration-200">

Mark All Present

</button>

<button onclick="saveAttendance()" class="bg-blue-600 hover:bg-blue-700 text-white px-4 py-2 rounded-lg font-medium transition-colors duration-200">

Save Attendance

</button>

</div>

</div>

</div>

<!-- Reports Tab -->

<div id="reports" class="tab-content">

<div class="mb-6">

<h2 class="text-3xl font-bold text-gray-800">Reports & Analytics</h2>

<p class="text-gray-600">Generate comprehensive reports and insights</p>

</div>

<!-- Report Cards -->

<div class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-6">

<div class="bg-white rounded-lg shadow-md p-6">

<div class="flex items-center justify-between mb-4">

<h3 class="text-lg font-semibold text-gray-800">Student Performance</h3>

<span class="text-2xl">📊</span>

</div>

<p class="text-gray-600 mb-4">Generate detailed performance reports for individual students or classes</p>

<button onclick="generateStudentReport()" class="w-full bg-blue-600 hover:bg-blue-700 text-white py-2 rounded-lg font-medium transition-colors duration-200">

Generate Report

</button>

</div>

<div class="bg-white rounded-lg shadow-md p-6">

<div class="flex items-center justify-between mb-4">

<h3 class="text-lg font-semibold text-gray-800">Attendance Summary</h3>

<span class="text-2xl">📅</span>

</div>

<p class="text-gray-600 mb-4">View attendance patterns and statistics across all courses</p>

<button onclick="generateAttendanceReport()" class="w-full bg-green-600 hover:bg-green-700 text-white py-2 rounded-lg font-medium transition-colors duration-200">

Generate Report

</button>

</div>

<div class="bg-white rounded-lg shadow-md p-6">

<div class="flex items-center justify-between mb-4">

<h3 class="text-lg font-semibold text-gray-800">Course Analytics</h3>

<span class="text-2xl">📈</span>

</div>

<p class="text-gray-600 mb-4">Analyze course performance and enrollment trends</p>

<button onclick="generateCourseReport()" class="w-full bg-purple-600 hover:bg-purple-700 text-white py-2 rounded-lg font-medium transition-colors duration-200">

Generate Report

</button>

</div>

</div>

<!-- Report Output -->

<div id="reportOutput" class="mt-8 bg-white rounded-lg shadow-md p-6" style="display: none;">

<h3 class="text-xl font-semibold text-gray-800 mb-4">Report Results</h3>

<div id="reportContent" class="space-y-4">

<!-- Report content will be displayed here -->

</div>

</div>

</div>

</main>

</div>

<!-- Modals -->

<!-- Add Student Modal -->

<div id="addStudentModal" class="fixed inset-0 bg-black bg-opacity-50 hidden items-center justify-center z-50">

<div class="bg-white rounded-lg p-6 w-full max-w-md mx-4">

<h3 class="text-xl font-semibold text-gray-800 mb-4">Add New Student</h3>

<form onsubmit="addStudent(event)">

<div class="space-y-4">

<div>

<label class="block text-sm font-medium text-gray-700 mb-2">Full Name</label>

<input type="text" id="studentName" required class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

</div>

<div>

<label class="block text-sm font-medium text-gray-700 mb-2">Email</label>

<input type="email" id="studentEmail" required class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

</div>

<div>

<label class="block text-sm font-medium text-gray-700 mb-2">Grade</label>

<select id="studentGrade" required class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

<option value="">Select Grade</option>

<option value="9">Grade 9</option>

<option value="10">Grade 10</option>

<option value="11">Grade 11</option>

<option value="12">Grade 12</option>

</select>

</div>

<div>

<label class="block text-sm font-medium text-gray-700 mb-2">Phone</label>

<input type="tel" id="studentPhone" class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

</div>

</div>

<div class="flex gap-4 mt-6">

<button type="button" onclick="hideAddStudentModal()" class="flex-1 bg-gray-300 hover:bg-gray-400 text-gray-700 py-2 rounded-lg font-medium transition-colors duration-200">

Cancel

</button>

<button type="submit" class="flex-1 bg-blue-600 hover:bg-blue-700 text-white py-2 rounded-lg font-medium transition-colors duration-200">

Add Student

</button>

</div>

</form>

</div>

</div>

<!-- Add Teacher Modal -->

<div id="addTeacherModal" class="fixed inset-0 bg-black bg-opacity-50 hidden items-center justify-center z-50">

<div class="bg-white rounded-lg p-6 w-full max-w-md mx-4">

<h3 class="text-xl font-semibold text-gray-800 mb-4">Add New Teacher</h3>

<form onsubmit="addTeacher(event)">

<div class="space-y-4">

<div>

<label class="block text-sm font-medium text-gray-700 mb-2">Full Name</label>

<input type="text" id="teacherName" required class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

</div>

<div>

<label class="block text-sm font-medium text-gray-700 mb-2">Email</label>

<input type="email" id="teacherEmail" required class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

</div>

<div>

<label class="block text-sm font-medium text-gray-700 mb-2">Subject</label>

<input type="text" id="teacherSubject" required class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

</div>

<div>

<label class="block text-sm font-medium text-gray-700 mb-2">Phone</label>

<input type="tel" id="teacherPhone" class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

</div>

</div>

<div class="flex gap-4 mt-6">

<button type="button" onclick="hideAddTeacherModal()" class="flex-1 bg-gray-300 hover:bg-gray-400 text-gray-700 py-2 rounded-lg font-medium transition-colors duration-200">

Cancel

</button>

<button type="submit" class="flex-1 bg-green-600 hover:bg-green-700 text-white py-2 rounded-lg font-medium transition-colors duration-200">

Add Teacher

</button>

</div>

</form>

</div>

</div>

<!-- Add Course Modal -->

<div id="addCourseModal" class="fixed inset-0 bg-black bg-opacity-50 hidden items-center justify-center z-50">

<div class="bg-white rounded-lg p-6 w-full max-w-md mx-4">

<h3 class="text-xl font-semibold text-gray-800 mb-4">Add New Course</h3>

<form onsubmit="addCourse(event)">

<div class="space-y-4">

<div>

<label class="block text-sm font-medium text-gray-700 mb-2">Course Name</label>

<input type="text" id="courseName" required class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

</div>

<div>

<label class="block text-sm font-medium text-gray-700 mb-2">Course Code</label>

<input type="text" id="courseCode" required class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

</div>

<div>

<label class="block text-sm font-medium text-gray-700 mb-2">Teacher</label>

<select id="courseTeacher" required class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

<option value="">Select Teacher</option>

</select>

</div>

<div>

<label class="block text-sm font-medium text-gray-700 mb-2">Credits</label>

<input type="number" id="courseCredits" min="1" max="6" required class="w-full px-4 py-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-blue-500 focus:border-transparent">

</div>

</div>

<div class="flex gap-4 mt-6">

<button type="button" onclick="hideAddCourseModal()" class="flex-1 bg-gray-300 hover:bg-gray-400 text-gray-700 py-2 rounded-lg font-medium transition-colors duration-200">

Cancel

</button>

<button type="submit" class="flex-1 bg-purple-600 hover:bg-purple-700 text-white py-2 rounded-lg font-medium transition-colors duration-200">

Add Course

</button>

</div>

</form>

</div>

</div>

<script>

// Data Storage

let students = [

{ id: 'STU001', name: 'Alice Johnson', email: 'alice.johnson@email.com', grade: '10', phone: '555-0101', status: 'Active' },

{ id: 'STU002', name: 'Bob Smith', email: 'bob.smith@email.com', grade: '11', phone: '555-0102', status: 'Active' },

{ id: 'STU003', name: 'Carol Davis', email: 'carol.davis@email.com', grade: '9', phone: '555-0103', status: 'Active' },

{ id: 'STU004', name: 'David Wilson', email: 'david.wilson@email.com', grade: '12', phone: '555-0104', status: 'Active' },

{ id: 'STU005', name: 'Emma Brown', email: 'emma.brown@email.com', grade: '10', phone: '555-0105', status: 'Active' }

];

let teachers = [

{ id: 'TEA001', name: 'Dr. Sarah Miller', email: 'sarah.miller@school.edu', subject: 'Mathematics', phone: '555-0201' },

{ id: 'TEA002', name: 'Prof. John Anderson', email: 'john.anderson@school.edu', subject: 'English Literature', phone: '555-0202' },

{ id: 'TEA003', name: 'Ms. Lisa Garcia', email: 'lisa.garcia@school.edu', subject: 'Science', phone: '555-0203' },

{ id: 'TEA004', name: 'Mr. Michael Chen', email: 'michael.chen@school.edu', subject: 'History', phone: '555-0204' }

];

let courses = [

{ id: 'COU001', name: 'Advanced Mathematics', code: 'MATH401', teacher: 'Dr. Sarah Miller', credits: 4, students: 25 },

{ id: 'COU002', name: 'English Literature', code: 'ENG301', teacher: 'Prof. John Anderson', credits: 3, students: 30 },

{ id: 'COU003', name: 'Physics', code: 'PHY201', teacher: 'Ms. Lisa Garcia', credits: 4, students: 22 },

{ id: 'COU004', name: 'World History', code: 'HIS101', teacher: 'Mr. Michael Chen', credits: 3, students: 28 }

];

let grades = [

{ studentId: 'STU001', studentName: 'Alice Johnson', courseId: 'COU001', courseName: 'Advanced Mathematics', grade: 92, date: '2024-01-15' },

{ studentId: 'STU002', studentName: 'Bob Smith', courseId: 'COU002', courseName: 'English Literature', grade: 88, date: '2024-01-15' },

{ studentId: 'STU003', studentName: 'Carol Davis', courseId: 'COU003', courseName: 'Physics', grade: 95, date: '2024-01-15' }

];

let attendance = {};

// Navigation Functions

function showTab(tabName) {

// Hide all tabs

document.querySelectorAll('.tab-content').forEach(tab => {

tab.classList.remove('active');

});

// Remove active class from all sidebar items

document.querySelectorAll('.sidebar-item').forEach(item => {

item.classList.remove('active');

});

// Show selected tab

document.getElementById(tabName).classList.add('active');

// Add active class to clicked sidebar item

event.target.classList.add('active');

// Load data for the selected tab

switch(tabName) {

case 'students':

loadStudents();

break;

case 'teachers':

loadTeachers();

break;

case 'courses':

loadCourses();

break;

case 'grades':

loadGrades();

populateGradeSelects();

break;

case 'attendance':

populateAttendanceSelects();

break;

}

}

// Student Management Functions

function loadStudents() {

const tbody = document.getElementById('studentsTableBody');

tbody.innerHTML = '';

students.forEach(student => {

const row = document.createElement('tr');

row.innerHTML = `

<td class="px-6 py-4 whitespace-nowrap">

<div class="flex items-center">

<div class="w-10 h-10 bg-blue-100 rounded-full flex items-center justify-center mr-3">

<span class="text-blue-600 font-semibold">${student.name.charAt(0)}</span>

</div>

<div>

<div class="text-sm font-medium text-gray-900">${student.name}</div>

</div>

</div>

</td>

<td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">${student.id}</td>

<td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">Grade ${student.grade}</td>

<td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">${student.email}</td>

<td class="px-6 py-4 whitespace-nowrap">

<span class="px-2 inline-flex text-xs leading-5 font-semibold rounded-full bg-green-100 text-green-800">

${student.status}

</span>

</td>

<td class="px-6 py-4 whitespace-nowrap text-sm font-medium">

<button onclick="editStudent('${student.id}')" class="text-blue-600 hover:text-blue-900 mr-3">Edit</button>

<button onclick="deleteStudent('${student.id}')" class="text-red-600 hover:text-red-900">Delete</button>

</td>

`;

tbody.appendChild(row);

});

}

function filterStudents() {

const searchTerm = document.getElementById('studentSearch').value.toLowerCase();

const gradeFilter = document.getElementById('gradeFilter').value;

const filteredStudents = students.filter(student => {

const matchesSearch = student.name.toLowerCase().includes(searchTerm) ||

student.email.toLowerCase().includes(searchTerm) ||

student.id.toLowerCase().includes(searchTerm);

const matchesGrade = !gradeFilter || student.grade === gradeFilter;

return matchesSearch && matchesGrade;

});

const tbody = document.getElementById('studentsTableBody');

tbody.innerHTML = '';

filteredStudents.forEach(student => {

const row = document.createElement('tr');

row.innerHTML = `

<td class="px-6 py-4 whitespace-nowrap">

<div class="flex items-center">

<div class="w-10 h-10 bg-blue-100 rounded-full flex items-center justify-center mr-3">

<span class="text-blue-600 font-semibold">${student.name.charAt(0)}</span>

</div>

<div>

<div class="text-sm font-medium text-gray-900">${student.name}</div>

</div>

</div>

</td>

<td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">${student.id}</td>

<td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">Grade ${student.grade}</td>

<td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">${student.email}</td>

<td class="px-6 py-4 whitespace-nowrap">

<span class="px-2 inline-flex text-xs leading-5 font-semibold rounded-full bg-green-100 text-green-800">

${student.status}

</span>

</td>

<td class="px-6 py-4 whitespace-nowrap text-sm font-medium">

<button onclick="editStudent('${student.id}')" class="text-blue-600 hover:text-blue-900 mr-3">Edit</button>

<button onclick="deleteStudent('${student.id}')" class="text-red-600 hover:text-red-900">Delete</button>

</td>

`;

tbody.appendChild(row);

});

}

function showAddStudentModal() {

document.getElementById('addStudentModal').classList.remove('hidden');

document.getElementById('addStudentModal').classList.add('flex');

}

function hideAddStudentModal() {

document.getElementById('addStudentModal').classList.add('hidden');

document.getElementById('addStudentModal').classList.remove('flex');

document.getElementById('addStudentModal').querySelector('form').reset();

}

function addStudent(event) {

event.preventDefault();

const name = document.getElementById('studentName').value;

const email = document.getElementById('studentEmail').value;

const grade = document.getElementById('studentGrade').value;

const phone = document.getElementById('studentPhone').value;

const newStudent = {

id: 'STU' + String(students.length + 1).padStart(3, '0'),

name: name,

email: email,

grade: grade,

phone: phone,

status: 'Active'

};

students.push(newStudent);

loadStudents();

hideAddStudentModal();

updateDashboardStats();

alert('Student added successfully!');

}

function editStudent(studentId) {

const student = students.find(s => s.id === studentId);

if (student) {

const newName = prompt('Enter new name:', student.name);

if (newName) {

student.name = newName;

loadStudents();

alert('Student updated successfully!');

}

}

}

function deleteStudent(studentId) {

if (confirm('Are you sure you want to delete this student?')) {

students = students.filter(s => s.id !== studentId);

loadStudents();

updateDashboardStats();

alert('Student deleted successfully!');

}

}

// Teacher Management Functions

function loadTeachers() {

const grid = document.getElementById('teachersGrid');

grid.innerHTML = '';

teachers.forEach(teacher => {

const card = document.createElement('div');

card.className = 'bg-white rounded-lg shadow-md p-6';

card.innerHTML = `

<div class="flex items-center mb-4">

<div class="w-12 h-12 bg-green-100 rounded-full flex items-center justify-center mr-4">

<span class="text-green-600 font-semibold text-lg">${teacher.name.charAt(0)}</span>

</div>

<div>

<h3 class="text-lg font-semibold text-gray-800">${teacher.name}</h3>

<p class="text-gray-600">${teacher.subject}</p>

</div>

</div>

<div class="space-y-2 text-sm text-gray-600">

<p><span class="font-medium">ID:</span> ${teacher.id}</p>

<p><span class="font-medium">Email:</span> ${teacher.email}</p>

<p><span class="font-medium">Phone:</span> ${teacher.phone}</p>

</div>

<div class="mt-4 flex gap-2">

<button onclick="editTeacher('${teacher.id}')" class="flex-1 bg-blue-600 hover:bg-blue-700 text-white py-2 rounded-lg font-medium transition-colors duration-200">

Edit

</button>

<button onclick="deleteTeacher('${teacher.id}')" class="flex-1 bg-red-600 hover:bg-red-700 text-white py-2 rounded-lg font-medium transition-colors duration-200">

Delete

</button>

</div>

`;

grid.appendChild(card);

});

}

function showAddTeacherModal() {

document.getElementById('addTeacherModal').classList.remove('hidden');

document.getElementById('addTeacherModal').classList.add('flex');

}

function hideAddTeacherModal() {

document.getElementById('addTeacherModal').classList.add('hidden');

document.getElementById('addTeacherModal').classList.remove('flex');

document.getElementById('addTeacherModal').querySelector('form').reset();

}

function addTeacher(event) {

event.preventDefault();

const name = document.getElementById('teacherName').value;

const email = document.getElementById('teacherEmail').value;

const subject = document.getElementById('teacherSubject').value;

const phone = document.getElementById('teacherPhone').value;

const newTeacher = {

id: 'TEA' + String(teachers.length + 1).padStart(3, '0'),

name: name,

email: email,

subject: subject,

phone: phone

};

teachers.push(newTeacher);

loadTeachers();

hideAddTeacherModal();

updateDashboardStats();

alert('Teacher added successfully!');

}

function editTeacher(teacherId) {

const teacher = teachers.find(t => t.id === teacherId);

if (teacher) {

const newName = prompt('Enter new name:', teacher.name);

if (newName) {

teacher.name = newName;

loadTeachers();

alert('Teacher updated successfully!');

}

}

}

function deleteTeacher(teacherId) {

if (confirm('Are you sure you want to delete this teacher?')) {

teachers = teachers.filter(t => t.id !== teacherId);

loadTeachers();

updateDashboardStats();

alert('Teacher deleted successfully!');

}

}

// Course Management Functions

function loadCourses() {

const grid = document.getElementById('coursesGrid');

grid.innerHTML = '';

courses.forEach(course => {

const card = document.createElement('div');

card.className = 'bg-white rounded-lg shadow-md p-6';

card.innerHTML = `

<div class="flex items-center justify-between mb-4">

<div class="w-12 h-12 bg-purple-100 rounded-lg flex items-center justify-center">

<span class="text-purple-600 text-xl">📖</span>

</div>

<span class="text-sm text-gray-500">${course.code}</span>

</div>

<h3 class="text-lg font-semibold text-gray-800 mb-2">${course.name}</h3>

<div class="space-y-2 text-sm text-gray-600 mb-4">

<p><span class="font-medium">Teacher:</span> ${course.teacher}</p>

<p><span class="font-medium">Credits:</span> ${course.credits}</p>

<p><span class="font-medium">Students:</span> ${course.students}</p>

</div>

<div class="flex gap-2">

<button onclick="editCourse('${course.id}')" class="flex-1 bg-blue-600 hover:bg-blue-700 text-white py-2 rounded-lg font-medium transition-colors duration-200">

Edit

</button>

<button onclick="deleteCourse('${course.id}')" class="flex-1 bg-red-600 hover:bg-red-700 text-white py-2 rounded-lg font-medium transition-colors duration-200">

Delete

</button>

</div>

`;

grid.appendChild(card);

});

}

function showAddCourseModal() {

// Populate teacher dropdown

const teacherSelect = document.getElementById('courseTeacher');

teacherSelect.innerHTML = '<option value="">Select Teacher</option>';

teachers.forEach(teacher => {

teacherSelect.innerHTML += `<option value="${teacher.name}">${teacher.name}</option>`;

});

document.getElementById('addCourseModal').classList.remove('hidden');

document.getElementById('addCourseModal').classList.add('flex');

}

function hideAddCourseModal() {

document.getElementById('addCourseModal').classList.add('hidden');

document.getElementById('addCourseModal').classList.remove('flex');

document.getElementById('addCourseModal').querySelector('form').reset();

}

function addCourse(event) {

event.preventDefault();

const name = document.getElementById('courseName').value;

const code = document.getElementById('courseCode').value;

const teacher = document.getElementById('courseTeacher').value;

const credits = document.getElementById('courseCredits').value;

const newCourse = {

id: 'COU' + String(courses.length + 1).padStart(3, '0'),

name: name,

code: code,

teacher: teacher,

credits: parseInt(credits),

students: 0

};

courses.push(newCourse);

loadCourses();

hideAddCourseModal();

updateDashboardStats();

alert('Course added successfully!');

}

function editCourse(courseId) {

const course = courses.find(c => c.id === courseId);

if (course) {

const newName = prompt('Enter new course name:', course.name);

if (newName) {

course.name = newName;

loadCourses();

alert('Course updated successfully!');

}

}

}

function deleteCourse(courseId) {

if (confirm('Are you sure you want to delete this course?')) {

courses = courses.filter(c => c.id !== courseId);

loadCourses();

updateDashboardStats();

alert('Course deleted successfully!');

}

}

// Grade Management Functions

function loadGrades() {

const tbody = document.getElementById('gradesTableBody');

tbody.innerHTML = '';

grades.forEach((grade, index) => {

const letterGrade = getLetterGrade(grade.grade);

const row = document.createElement('tr');

row.innerHTML = `

<td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">${grade.studentName}</td>

<td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">${grade.courseName}</td>

<td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">${grade.grade}%</td>

<td class="px-6 py-4 whitespace-nowrap">

<span class="px-2 inline-flex text-xs leading-5 font-semibold rounded-full ${getGradeColor(letterGrade)}">

${letterGrade}

</span>

</td>

<td class="px-6 py-4 whitespace-nowrap text-sm text-gray-900">${grade.date}</td>

<td class="px-6 py-4 whitespace-nowrap text-sm font-medium">

<button onclick="editGrade(${index})" class="text-blue-600 hover:text-blue-900 mr-3">Edit</button>

<button onclick="deleteGrade(${index})" class="text-red-600 hover:text-red-900">Delete</button>

</td>

`;

tbody.appendChild(row);

});

}

function populateGradeSelects() {

const studentSelect = document.getElementById('gradeStudent');

const courseSelect = document.getElementById('gradeCourse');

studentSelect.innerHTML = '<option value="">Select Student</option>';

students.forEach(student => {

studentSelect.innerHTML += `<option value="${student.id}">${student.name}</option>`;

});

courseSelect.innerHTML = '<option value="">Select Course</option>';

courses.forEach(course => {

courseSelect.innerHTML += `<option value="${course.id}">${course.name}</option>`;

});

}

function addGrade() {

const studentId = document.getElementById('gradeStudent').value;

const courseId = document.getElementById('gradeCourse').value;

const gradeValue = document.getElementById('gradeValue').value;

if (!studentId || !courseId || !gradeValue) {

alert('Please fill in all fields');

return;

}

const student = students.find(s => s.id === studentId);

const course = courses.find(c => c.id === courseId);

const newGrade = {

studentId: studentId,

studentName: student.name,

courseId: courseId,

courseName: course.name,

grade: parseInt(gradeValue),

date: new Date().toISOString().split('T')[0]

};

grades.push(newGrade);

loadGrades();

// Clear form

document.getElementById('gradeStudent').value = '';

document.getElementById('gradeCourse').value = '';

document.getElementById('gradeValue').value = '';

alert('Grade added successfully!');

}

function getLetterGrade(numericGrade) {

if (numericGrade >= 90) return 'A';

if (numericGrade >= 80) return 'B';

if (numericGrade >= 70) return 'C';

if (numericGrade >= 60) return 'D';

return 'F';

}

function getGradeColor(letterGrade) {

switch(letterGrade) {

case 'A': return 'bg-green-100 text-green-800';

case 'B': return 'bg-blue-100 text-blue-800';

case 'C': return 'bg-yellow-100 text-yellow-800';

case 'D': return 'bg-orange-100 text-orange-800';

case 'F': return 'bg-red-100 text-red-800';

default: return 'bg-gray-100 text-gray-800';

}

}

function editGrade(index) {

const grade = grades[index];

const newGrade = prompt('Enter new grade (0-100):', grade.grade);

if (newGrade && !isNaN(newGrade) && newGrade >= 0 && newGrade <= 100) {

grades[index].grade = parseInt(newGrade);

loadGrades();

alert('Grade updated successfully!');

}

}

function deleteGrade(index) {

if (confirm('Are you sure you want to delete this grade?')) {

grades.splice(index, 1);

loadGrades();

alert('Grade deleted successfully!');

}

}

// Attendance Management Functions

function populateAttendanceSelects() {

const courseSelect = document.getElementById('attendanceCourse');

courseSelect.innerHTML = '<option value="">Select Course</option>';

courses.forEach(course => {

courseSelect.innerHTML += `<option value="${course.id}">${course.name}</option>`;

});

// Set today's date

document.getElementById('attendanceDate').value = new Date().toISOString().split('T')[0];

}

function loadAttendance() {

const date = document.getElementById('attendanceDate').value;

const courseId = document.getElementById('attendanceCourse').value;

if (!date || !courseId) {

alert('Please select both date and course');

return;

}

const course = courses.find(c => c.id === courseId);

const attendanceKey = `${date}-${courseId}`;

const studentsContainer = document.getElementById('attendanceStudents');

studentsContainer.innerHTML = `

<h4 class="font-semibold text-gray-800 mb-3">Attendance for ${course.name} - ${date}</h4>

<div class="space-y-3">

`;

students.forEach(student => {

const isPresent = attendance[attendanceKey] && attendance[attendanceKey][student.id] === 'present';

studentsContainer.innerHTML += `

<div class="flex items-center justify-between p-3 bg-gray-50 rounded-lg">

<div class="flex items-center">

<div class="w-8 h-8 bg-blue-100 rounded-full flex items-center justify-center mr-3">

<span class="text-blue-600 font-semibold text-sm">${student.name.charAt(0)}</span>

</div>

<span class="font-medium text-gray-800">${student.name}</span>

</div>

<div class="flex gap-2">

<label class="flex items-center">

<input type="radio" name="attendance-${student.id}" value="present" ${isPresent ? 'checked' : ''} class="mr-2">

<span class="text-green-600">Present</span>

</label>

<label class="flex items-center">

<input type="radio" name="attendance-${student.id}" value="absent" ${!isPresent && attendance[attendanceKey] && attendance[attendanceKey][student.id] ? 'checked' : ''} class="mr-2">

<span class="text-red-600">Absent</span>

</label>

</div>

</div>

`;

});

studentsContainer.innerHTML += '</div>';

}

function markAllPresent() {

const radios = document.querySelectorAll('input[type="radio"][value="present"]');

radios.forEach(radio => {

radio.checked = true;

});

}

function saveAttendance() {

const date = document.getElementById('attendanceDate').value;

const courseId = document.getElementById('attendanceCourse').value;

if (!date || !courseId) {

alert('Please select both date and course');

return;

}

const attendanceKey = `${date}-${courseId}`;

attendance[attendanceKey] = {};

students.forEach(student => {

const presentRadio = document.querySelector(`input[name="attendance-${student.id}"][value="present"]`);

const absentRadio = document.querySelector(`input[name="attendance-${student.id}"][value="absent"]`);

if (presentRadio && presentRadio.checked) {

attendance[attendanceKey][student.id] = 'present';

} else if (absentRadio && absentRadio.checked) {

attendance[attendanceKey][student.id] = 'absent';

}

});

alert('Attendance saved successfully!');

}

// Report Functions

function generateStudentReport() {

const reportOutput = document.getElementById('reportOutput');

const reportContent = document.getElementById('reportContent');

let reportHtml = '<h4 class="text-lg font-semibold text-gray-800 mb-4">Student Performance Report</h4>';

students.forEach(student => {

const studentGrades = grades.filter(g => g.studentId === student.id);

const avgGrade = studentGrades.length > 0 ?

(studentGrades.reduce((sum, g) => sum + g.grade, 0) / studentGrades.length).toFixed(1) : 'N/A';

reportHtml += `

<div class="bg-gray-50 p-4 rounded-lg mb-3">

<h5 class="font-semibold text-gray-800">${student.name} (${student.id})</h5>

<p class="text-gray-600">Grade: ${student.grade} | Average Score: ${avgGrade}% | Courses: ${studentGrades.length}</p>

</div>

`;

});

reportContent.innerHTML = reportHtml;

reportOutput.style.display = 'block';

}

function generateAttendanceReport() {

const reportOutput = document.getElementById('reportOutput');

const reportContent = document.getElementById('reportContent');

let reportHtml = '<h4 class="text-lg font-semibold text-gray-800 mb-4">Attendance Summary Report</h4>';

const totalRecords = Object.keys(attendance).length;

let presentCount = 0;

let totalCount = 0;

Object.values(attendance).forEach(dayAttendance => {

Object.values(dayAttendance).forEach(status => {

totalCount++;

if (status === 'present') presentCount++;

});

});

const attendanceRate = totalCount > 0 ? ((presentCount / totalCount) \* 100).toFixed(1) : 0;

reportHtml += `

<div class="bg-gray-50 p-4 rounded-lg mb-3">

<h5 class="font-semibold text-gray-800">Overall Statistics</h5>

<p class="text-gray-600">Total Attendance Records: ${totalRecords}</p>

<p class="text-gray-600">Overall Attendance Rate: ${attendanceRate}%</p>

<p class="text-gray-600">Present: ${presentCount} | Absent: ${totalCount - presentCount}</p>

</div>

`;

reportContent.innerHTML = reportHtml;

reportOutput.style.display = 'block';

}

function generateCourseReport() {

const reportOutput = document.getElementById('reportOutput');

const reportContent = document.getElementById('reportContent');

let reportHtml = '<h4 class="text-lg font-semibold text-gray-800 mb-4">Course Analytics Report</h4>';

courses.forEach(course => {

const courseGrades = grades.filter(g => g.courseId === course.id);

const avgGrade = courseGrades.length > 0 ?

(courseGrades.reduce((sum, g) => sum + g.grade, 0) / courseGrades.length).toFixed(1) : 'N/A';

reportHtml += `

<div class="bg-gray-50 p-4 rounded-lg mb-3">

<h5 class="font-semibold text-gray-800">${course.name} (${course.code})</h5>

<p class="text-gray-600">Teacher: ${course.teacher} | Credits: ${course.credits}</p>

<p class="text-gray-600">Enrolled Students: ${course.students} | Average Grade: ${avgGrade}%</p>

<p class="text-gray-600">Graded Assignments: ${courseGrades.length}</p>

</div>

`;

});

reportContent.innerHTML = reportHtml;

reportOutput.style.display = 'block';

}

// Dashboard Functions

function updateDashboardStats() {

document.getElementById('totalStudents').textContent = students.length.toLocaleString();

document.getElementById('totalTeachers').textContent = teachers.length;

document.getElementById('totalCourses').textContent = courses.length;

}

// Initialize the application

document.addEventListener('DOMContentLoaded', function() {

// Set today's date for attendance

document.getElementById('attendanceDate').value = new Date().toISOString().split('T')[0];

// Load initial data

loadStudents();

updateDashboardStats();

});

</script>

<script>(function(){function c(){var b=a.contentDocument||a.contentWindow.document;if(b){var d=b.createElement('script');d.innerHTML="window.\_\_CF$cv$params={r:'967bf95521fb2695',t:'MTc1Mzk1MzE0NS4wMDAwMDA='};var a=document.createElement('script');a.nonce='';a.src='/cdn-cgi/challenge-platform/scripts/jsd/main.js';document.getElementsByTagName('head')[0].appendChild(a);";b.getElementsByTagName('head')[0].appendChild(d)}}if(document.body){var a=document.createElement('iframe');a.height=1;a.width=1;a.style.position='absolute';a.style.top=0;a.style.left=0;a.style.border='none';a.style.visibility='hidden';document.body.appendChild(a);if('loading'!==document.readyState)c();else if(window.addEventListener)document.addEventListener('DOMContentLoaded',c);else{var e=document.onreadystatechange||function(){};document.onreadystatechange=function(b){e(b);'loading'!==document.readyState&&(document.onreadystatechange=e,c())}}}})();</script></body>

</html>