<!DOCTYPE html>

<html lang="th">

<head>

    <meta charset="UTF-8">

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

    <title>ระบบจัดการสต็อกวัสดุ</title>

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

    <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

    <link rel="preconnect" href="https://fonts.googleapis.com">

    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

    <link href="https://fonts.googleapis.com/css2?family=Sarabun:wght@400;500;700&display=swap" rel="stylesheet">

    <!-- Chosen Palette: Warm Neutrals -->

    <!-- Application Structure Plan: The application uses a tabbed navigation interface (Dashboard, Receive, Request, History, Summary) to mirror the task-oriented nature of the source Google Sheets report. This structure separates data visualization (Dashboard & Summary) from data entry (Forms), which is a standard and intuitive UX pattern. The dashboard provides an at-a-glance overview with a sortable inventory table and a summary chart, while dedicated forms streamline the process of recording transactions. The new summary report view allows for filtered stock analysis over a user-defined date range, which is critical for accounting purposes like GL reconciliation. A unified, filterable history log enhances audit capabilities. This design was chosen to transform the static, multi-sheet report into a dynamic, single-page, task-focused web application. -->

    <!-- Visualization & Content Choices:

        - Inventory List: Report's 'ฐานข้อมูลวัสดุ' -> Goal: Inform/Compare -> Method: Sortable HTML Table -> Interaction: Click headers to sort, search bar to filter -> Justification: Best for structured data, familiar to users.

        - Stock Levels: Calculated from 'ใบรับ/เบิก' -> Goal: Compare -> Method: Chart.js Bar Chart -> Interaction: Hover for tooltips -> Justification: Quick visual comparison of quantities.

        - Data Entry: 'ใบรับ/เบิก' sheets -> Goal: Input Data -> Method: HTML Forms -> Interaction: Dropdown for material selection auto-populates details -> Justification: Standard for input, auto-population minimizes error (mimics VLOOKUP).

        - Transaction Log: All 'ใบรับ/เบิก' entries -> Goal: Review/Audit -> Method: Filterable HTML Table -> Interaction: Dropdowns to filter by type/material -> Justification: More powerful and flexible than separate static lists.

        - Summary Report: A new feature to calculate stock over a specific date range -> Goal: Reconciliation with GL -> Method: Filterable HTML Table with date pickers -> Interaction: Select dates and click "Generate Report" -> Justification: Directly addresses the user's need for a GL-focused report.

        - Library/Method: Chart.js for charts, Vanilla JS for all logic, Tailwind CSS for styling. -->

    <!-- CONFIRMATION: NO SVG graphics used. NO Mermaid JS used. -->

    <style>

        body {

            font-family: 'Sarabun', sans-serif;

            background-color: #FDFBF7;

            color: #4A4A4A;

        }

        .nav-btn {

            transition: all 0.3s ease;

        }

        .nav-btn.active {

            background-color: #D4A373;

            color: #FFFFFF;

            box-shadow: 0 2px 4px rgba(0,0,0,0.1);

        }

        .nav-btn:not(.active):hover {

            background-color: #EFEBE5;

        }

        .table-header-sortable {

            cursor: pointer;

            user-select: none;

        }

        .table-header-sortable:hover {

            background-color: #f0ebe5;

        }

        .chart-container {

            position: relative;

            width: 100%;

            max-width: 800px;

            margin-left: auto;

            margin-right: auto;

            height: 400px;

            max-height: 50vh;

        }

        .low-stock {

            background-color: #FEF2F2;

        }

        .low-stock:hover {

            background-color: #FEE2E2;

        }

    </style>

</head>

<body class="antialiased">

    <div class="container mx-auto p-4 sm:p-6 lg:p-8">

        <header class="text-center mb-8">

            <h1 class="text-3xl md:text-4xl font-bold text-[#3D405B]">ระบบจัดการสต็อกวัสดุ</h1>

            <p class="text-gray-600 mt-2">แอปพลิเคชันสำหรับบริหารจัดการการเบิก-จ่ายวัสดุคงคลัง</p>

        </header>

        <nav class="flex justify-center flex-wrap gap-2 mb-8">

            <button id="nav-dashboard" class="nav-btn active px-4 py-2 rounded-lg font-semibold" onclick="switchView('dashboard')">ภาพรวม</button>

            <button id="nav-receipt" class="nav-btn px-4 py-2 rounded-lg font-semibold" onclick="switchView('receipt')">รับวัสดุ</button>

            <button id="nav-request" class="nav-btn px-4 py-2 rounded-lg font-semibold" onclick="switchView('request')">เบิกวัสดุ</button>

            <button id="nav-summary" class="nav-btn px-4 py-2 rounded-lg font-semibold" onclick="switchView('summary')">รายงานสรุป</button>

            <button id="nav-history" class="nav-btn px-4 py-2 rounded-lg font-semibold" onclick="switchView('history')">ประวัติธุรกรรม</button>

        </nav>

        <main>

            <!-- Dashboard View -->

            <div id="view-dashboard">

                <section class="bg-white p-6 rounded-xl shadow-md mb-8">

                    <h2 class="text-2xl font-bold mb-4 text-[#3D405B]">ภาพรวมสต็อก</h2>

                    <p class="text-gray-600 mb-6">สรุปข้อมูลวัสดุคงคลังทั้งหมด คุณสามารถค้นหาและจัดเรียงข้อมูลในตารางได้</p>

                    <div class="grid grid-cols-1 md:grid-cols-3 gap-4 mb-6 text-center">

                        <div class="bg-[#EFEBE5] p-4 rounded-lg">

                            <h3 class="font-semibold text-gray-700">จำนวนรายการทั้งหมด</h3>

                            <p id="total-items" class="text-2xl font-bold text-[#D4A373]">0</p>

                        </div>

                        <div class="bg-[#EFEBE5] p-4 rounded-lg">

                            <h3 class="font-semibold text-gray-700">มูลค่าสต็อกรวม</h3>

                            <p id="total-value" class="text-2xl font-bold text-[#D4A373]">฿0.00</p>

                        </div>

                        <div class="bg-[#EFEBE5] p-4 rounded-lg">

                            <h3 class="font-semibold text-gray-700">รายการที่ต้องสั่งซื้อเพิ่ม</h3>

                            <p id="low-stock-count" class="text-2xl font-bold text-[#D4A373]">0</p>

                        </div>

                    </div>

                    <div class="mb-4">

                        <input type="text" id="search-inventory" placeholder="ค้นหารายการวัสดุ..." class="w-full p-2 border border-gray-300 rounded-lg focus:ring-2 focus:ring-[#D4A373] focus:border-transparent">

                    </div>

                    <div class="overflow-x-auto">

                        <table class="w-full text-left">

                            <thead class="bg-gray-100">

                                <tr>

                                    <th class="p-3 table-header-sortable" onclick="sortTable('inventory-table', 0)">รหัสวัสดุ</th>

                                    <th class="p-3 table-header-sortable" onclick="sortTable('inventory-table', 1)">รายละเอียด</th>

                                    <th class="p-3 text-right table-header-sortable" onclick="sortTable('inventory-table', 2, true)">ราคา/หน่วย</th>

                                    <th class="p-3 text-right table-header-sortable" onclick="sortTable('inventory-table', 3, true)">ยอดคงเหลือ</th>

                                    <th class="p-3">หน่วยนับ</th>

                                    <th class="p-3 text-right table-header-sortable" onclick="sortTable('inventory-table', 5, true)">มูลค่ารวม</th>

                                </tr>

                            </thead>

                            <tbody id="inventory-table">

                                <!-- Data will be populated by JS -->

                            </tbody>

                        </table>

                    </div>

                </section>

                <section class="bg-white p-6 rounded-xl shadow-md">

                    <h2 class="text-2xl font-bold mb-4 text-[#3D405B]">ปริมาณวัสดุในสต็อก</h2>

                     <p class="text-gray-600 mb-6">แผนภูมินี้แสดงปริมาณวัสดุ 10 อันดับแรกที่มีจำนวนคงเหลือมากที่สุด</p>

                    <div class="chart-container">

                        <canvas id="inventoryChart"></canvas>

                    </div>

                </section>

            </div>

            <!-- Receipt Form View -->

            <div id="view-receipt" class="hidden">

                <section class="bg-white p-6 rounded-xl shadow-md max-w-3xl mx-auto">

                    <h2 class="text-2xl font-bold mb-4 text-[#3D405B]">บันทึกการรับวัสดุ</h2>

                    <p class="text-gray-600 mb-6">กรอกฟอร์มด้านล่างเพื่อบันทึกข้อมูลการรับวัสดุเข้าสู่คลังสินค้า</p>

                    <form id="receipt-form" class="space-y-4">

                        <div class="grid grid-cols-1 md:grid-cols-2 gap-4">

                            <input type="date" name="date" class="p-2 border rounded-lg w-full" required>

                            <input type="text" name="company" placeholder="บริษัท" class="p-2 border rounded-lg w-full" required>

                        </div>

                        <div class="grid grid-cols-1 md:grid-cols-2 gap-4">

                            <input type="text" name="po" placeholder="ใบสั่งซื้อ (PO)" class="p-2 border rounded-lg w-full" required>

                             <select name="materialId" id="receipt-material-select" class="p-2 border rounded-lg w-full" required>

                                <option value="">-- เลือกรหัสวัสดุ --</option>

                            </select>

                        </div>

                         <div class="grid grid-cols-1 md:grid-cols-2 gap-4">

                            <div class="p-2 border rounded-lg w-full bg-gray-100 text-gray-700">

                                คงเหลือ: <span id="receipt-current-stock" class="font-bold">0</span>

                            </div>

                            <input type="number" name="quantity" placeholder="จำนวน" min="1" class="p-2 border rounded-lg w-full" required>

                        </div>

                        <div class="grid grid-cols-1 md:grid-cols-2 gap-4">

                            <input type="number" name="price" placeholder="ราคา/หน่วย" step="0.01" min="0" class="p-2 border rounded-lg w-full" required>

                            <input type="text" name="project" placeholder="โครงการ (ถ้ามี)" class="p-2 border rounded-lg w-full">

                        </div>

                        <div class="grid grid-cols-1 md:grid-cols-2 gap-4">

                            <input type="text" name="ref" placeholder="เอกสารอ้างอิง (ถ้ามี)" class="p-2 border rounded-lg w-full">

                        </div>

                        <button type="submit" class="w-full bg-[#D4A373] text-white font-bold py-3 px-4 rounded-lg hover:bg-[#c8935a] transition-colors">บันทึกการรับ</button>

                    </form>

                </section>

            </div>

            <!-- Request Form View -->

            <div id="view-request" class="hidden">

                 <section class="bg-white p-6 rounded-xl shadow-md max-w-3xl mx-auto">

                    <h2 class="text-2xl font-bold mb-4 text-[#3D405B]">บันทึกการเบิกวัสดุ</h2>

                    <p class="text-gray-600 mb-6">กรอกฟอร์มด้านล่างเพื่อบันทึกข้อมูลการเบิกวัสดุออกจากคลังสินค้า</p>

                    <form id="request-form" class="space-y-4">

                        <div class="grid grid-cols-1 md:grid-cols-2 gap-4">

                            <input type="date" name="date" class="p-2 border rounded-lg w-full" required>

                            <input type="text" name="requester" placeholder="ผู้เบิก" class="p-2 border rounded-lg w-full" required>

                        </div>

                        <div class="grid grid-cols-1 md:grid-cols-2 gap-4">

                           <select name="materialId" id="request-material-select" class="p-2 border rounded-lg w-full" required>

                                <option value="">-- เลือกรหัสวัสดุ --</option>

                            </select>

                            <div class="p-2 border rounded-lg w-full bg-gray-100 text-gray-700">

                                คงเหลือ: <span id="request-current-stock" class="font-bold">0</span>

                            </div>

                        </div>

                         <div class="grid grid-cols-1 md:grid-cols-2 gap-4">

                            <input type="number" name="quantity" placeholder="จำนวน" min="1" class="p-2 border rounded-lg w-full" required>

                            <input type="text" name="project" placeholder="โครงการ (ถ้ามี)" class="p-2 border rounded-lg w-full">

                        </div>

                        <div class="grid grid-cols-1 md:grid-cols-2 gap-4">

                            <input type="text" name="ref" placeholder="เอกสารอ้างอิง (ถ้ามี)" class="p-2 border rounded-lg w-full">

                        </div>

                        <button type="submit" class="w-full bg-[#D4A373] text-white font-bold py-3 px-4 rounded-lg hover:bg-[#c8935a] transition-colors">บันทึกการเบิก</button>

                    </form>

                </section>

            </div>

            <!-- Summary Report View -->

            <div id="view-summary" class="hidden">

                <section class="bg-white p-6 rounded-xl shadow-md">

                    <h2 class="text-2xl font-bold mb-4 text-[#3D405B]">รายงานสรุปยอดคงเหลือ</h2>

                    <p class="text-gray-600 mb-6">เลือกช่วงเวลาเพื่อดูยอดคงเหลือของวัสดุทั้งหมดในช่วงนั้น</p>

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

                        <div>

                            <label for="report-from-date" class="block text-sm font-medium text-gray-700">จากวันที่</label>

                            <input type="date" id="report-from-date" class="mt-1 p-2 border rounded-lg w-full">

                        </div>

                        <div>

                            <label for="report-to-date" class="block text-sm font-medium text-gray-700">ถึงวันที่</label>

                            <input type="date" id="report-to-date" class="mt-1 p-2 border rounded-lg w-full">

                        </div>

                        <button id="generate-report-btn" class="w-full bg-[#D4A373] text-white font-bold py-3 px-4 rounded-lg hover:bg-[#c8935a] transition-colors">สร้างรายงาน</button>

                    </div>

                    <div class="overflow-x-auto">

                        <table class="w-full text-left">

                            <thead class="bg-gray-100">

                                <tr>

                                    <th class="p-3">รหัสวัสดุ</th>

                                    <th class="p-3">รายละเอียด</th>

                                    <th class="p-3 text-right">ราคา/หน่วย</th>

                                    <th class="p-3 text-right">ยอดคงเหลือ</th>

                                    <th class="p-3">หน่วยนับ</th>

                                    <th class="p-3 text-right">มูลค่ารวม</th>

                                </tr>

                            </thead>

                            <tbody id="summary-table">

                                <!-- Report data will be populated by JS -->

                            </tbody>

                        </table>

                    </div>

                </section>

            </div>

            <!-- History View -->

            <div id="view-history" class="hidden">

                <section class="bg-white p-6 rounded-xl shadow-md">

                    <h2 class="text-2xl font-bold mb-4 text-[#3D405B]">ประวัติธุรกรรม</h2>

                    <p class="text-gray-600 mb-6">แสดงประวัติการรับและเบิกวัสดุทั้งหมด คุณสามารถกรองข้อมูลเพื่อดูเฉพาะรายการที่สนใจได้</p>

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

                        <select id="filter-type" class="p-2 border rounded-lg w-full">

                            <option value="all">ธุรกรรมทั้งหมด</option>

                            <option value="receipt">การรับ</option>

                            <option value="request">การเบิก</option>

                        </select>

                        <select id="filter-material" class="p-2 border rounded-lg w-full">

                            <option value="all">วัสดุทั้งหมด</option>

                        </select>

                        <input type="month" id="filter-month" class="p-2 border rounded-lg w-full">

                    </div>

                    <div class="overflow-x-auto">

                        <table class="w-full text-left">

                            <thead class="bg-gray-100">

                                <tr>

                                    <th class="p-3">วันที่</th>

                                    <th class="p-3">ประเภท</th>

                                    <th class="p-3">รายละเอียด</th>

                                    <th class="p-3 text-right">จำนวน</th>

                                    <th class="p-3">โครงการ</th>

                                    <th class="p-3">อ้างอิง</th>

                                </tr>

                            </thead>

                            <tbody id="history-table">

                                <!-- Data will be populated by JS -->

                            </tbody>

                        </table>

                    </div>

                </section>

            </div>

        </main>

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

            <div class="bg-white p-6 rounded-lg shadow-xl max-w-sm w-full text-center">

                <h3 id="alert-title" class="text-xl font-bold mb-2"></h3>

                <p id="alert-message" class="text-gray-700 mb-4"></p>

                <button onclick="closeAlert()" class="bg-[#D4A373] text-white px-4 py-2 rounded-lg">ตกลง</button>

            </div>

        </div>

    </div>

    <script>

        const LOW\_STOCK\_THRESHOLD = 10;

        const initialData = {

            materials: [

                { id: 'MAT-001', description: 'กระดาษ A4', price: 150, unit: 'รีม' },

                { id: 'MAT-002', description: 'ปากกาลูกลื่นสีน้ำเงิน', price: 10, unit: 'ด้าม' },

                { id: 'MAT-003', description: 'แฟ้มเอกสาร', price: 30, unit: 'อัน' },

                { id: 'MAT-004', description: 'คลิปหนีบกระดาษ', price: 20, unit: 'กล่อง' },

                { id: 'MAT-005', description: 'หมึกพิมพ์ดำ', price: 850, unit: 'ตลับ' },

            ],

            transactions: [

                { type: 'receipt', date: '2025-08-01', company: 'บจก.เครื่องเขียนไทย', po: 'PO-001', materialId: 'MAT-001', quantity: 100, price: 150, project: '-', ref: 'INV-001' },

                { type: 'receipt', date: '2025-08-05', company: 'บจก.อุปกรณ์สำนักงาน', po: 'PO-002', materialId: 'MAT-002', quantity: 50, price: 10, project: 'โครงการ ABC', ref: 'INV-002' },

                { type: 'request', date: '2025-08-02', requester: 'สมชาย', materialId: 'MAT-001', quantity: 5, project: 'โครงการ ABC', ref: 'DR-001' },

                { type: 'request', date: '2025-08-06', requester: 'มารีย์', materialId: 'MAT-002', quantity: 10, project: 'โครงการ XYZ', ref: 'DR-002' },

                { type: 'receipt', date: '2025-08-07', company: 'บจก.เครื่องเขียนไทย', po: 'PO-003', materialId: 'MAT-003', quantity: 20, price: 30, project: '-', ref: 'INV-003' },

                { type: 'receipt', date: '2025-08-08', company: 'ซัพพลายเออร์', po: 'PO-004', materialId: 'MAT-004', quantity: 5, price: 20, project: '-', ref: 'INV-004' },

                { type: 'request', date: '2025-08-09', requester: 'สมศรี', materialId: 'MAT-004', quantity: 2, project: 'การตลาด', ref: 'DR-003' },

            ]

        };

        let materials = [...initialData.materials];

        let transactions = [...initialData.transactions];

        let inventoryChart;

        function calculateStock(materialId, transactionList = transactions) {

            try {

                const receipts = transactionList

                    .filter(t => t.type === 'receipt' && t.materialId === materialId)

                    .reduce((sum, t) => sum + t.quantity, 0);

                const requests = transactionList

                    .filter(t => t.type === 'request' && t.materialId === materialId)

                    .reduce((sum, t) => sum + t.quantity, 0);

                return receipts - requests;

            } catch (error) {

                console.error(`Error calculating stock for ${materialId}:`, error);

                return 0;

            }

        }

        function showAlert(title, message) {

            document.getElementById('alert-title').innerText = title;

            document.getElementById('alert-message').innerText = message;

            document.getElementById('alert-modal').classList.remove('hidden');

        }

        function closeAlert() {

            document.getElementById('alert-modal').classList.add('hidden');

        }

        function updateStockDisplay(formId, materialId) {

            const stockDisplay = document.getElementById(`${formId}-current-stock`);

            if (stockDisplay) {

                if (materialId) {

                    const stock = calculateStock(materialId);

                    stockDisplay.innerText = stock.toString();

                    if (stock <= LOW\_STOCK\_THRESHOLD) {

                        stockDisplay.classList.add('text-red-600');

                        stockDisplay.classList.remove('text-green-600');

                    } else {

                        stockDisplay.classList.add('text-green-600');

                        stockDisplay.classList.remove('text-red-600');

                    }

                } else {

                    stockDisplay.innerText = '0';

                    stockDisplay.classList.remove('text-red-600', 'text-green-600');

                }

            }

        }

        function renderInventoryTable(filteredMaterials) {

            try {

                const tableBody = document.getElementById('inventory-table');

                const materialsToRender = filteredMaterials || materials;

                tableBody.innerHTML = '';

                let totalValue = 0;

                let lowStockCount = 0;

                const inventoryData = materialsToRender.map(m => {

                    const stock = calculateStock(m.id);

                    const value = stock \* m.price;

                    return { ...m, stock, value };

                });

                inventoryData.forEach(m => {

                    const isLowStock = m.stock <= LOW\_STOCK\_THRESHOLD && m.stock > 0;

                    if (isLowStock) lowStockCount++;

                    totalValue += m.value;

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

                    row.className = `border-b border-gray-200 hover:bg-gray-50 ${isLowStock ? 'low-stock' : ''}`;

                    row.innerHTML = `

                        <td class="p-3">${m.id}</td>

                        <td class="p-3 font-medium">${m.description}</td>

                        <td class="p-3 text-right">${m.price.toFixed(2)}</td>

                        <td class="p-3 text-right font-bold ${m.stock <= LOW\_STOCK\_THRESHOLD ? 'text-red-600' : 'text-green-600'}">${m.stock}</td>

                        <td class="p-3">${m.unit}</td>

                        <td class="p-3 text-right">${m.value.toFixed(2)}</td>

                    `;

                    tableBody.appendChild(row);

                });

                document.getElementById('total-items').innerText = materials.length;

                document.getElementById('total-value').innerText = `฿${totalValue.toLocaleString('th-TH', { minimumFractionDigits: 2, maximumFractionDigits: 2 })}`;

                document.getElementById('low-stock-count').innerText = lowStockCount;

            } catch (error) {

                console.error('Error rendering inventory table:', error);

            }

        }

        function renderInventoryChart() {

            try {

                const ctx = document.getElementById('inventoryChart').getContext('2d');

                const inventoryData = materials

                    .map(m => ({

                        description: m.description,

                        stock: calculateStock(m.id)

                    }))

                    .filter(d => d.stock > 0)

                    .sort((a, b) => b.stock - a.stock)

                    .slice(0, 10);

                if (inventoryChart) {

                    inventoryChart.destroy();

                }

                inventoryChart = new Chart(ctx, {

                    type: 'bar',

                    data: {

                        labels: inventoryData.map(d => d.description),

                        datasets: [{

                            label: 'ยอดคงเหลือ',

                            data: inventoryData.map(d => d.stock),

                            backgroundColor: '#D4A373',

                            borderColor: '#c8935a',

                            borderWidth: 1

                        }]

                    },

                    options: {

                        responsive: true,

                        maintainAspectRatio: false,

                        scales: {

                            y: {

                                beginAtZero: true

                            },

                            x: {

                                ticks: {

                                    callback: function(value, index, values) {

                                        const label = this.getLabelForValue(value);

                                        return label.length > 16 ? label.substring(0, 16) + '...' : label;

                                    }

                                }

                            }

                        },

                        plugins: {

                            legend: {

                                display: false

                            },

                            tooltip: {

                                callbacks: {

                                    label: function(context) {

                                        return `จำนวน: ${context.parsed.y}`;

                                    }

                                }

                            }

                        }

                    }

                });

            } catch (error) {

                console.error('Error rendering inventory chart:', error);

            }

        }

        function renderSummaryReport() {

            try {

                const tableBody = document.getElementById('summary-table');

                tableBody.innerHTML = '';

                const fromDateStr = document.getElementById('report-from-date').value;

                const toDateStr = document.getElementById('report-to-date').value;

                if (!fromDateStr || !toDateStr) {

                    showAlert('ข้อผิดพลาด', 'กรุณาเลือกช่วงเวลาให้ครบถ้วน');

                    return;

                }

                const fromDate = new Date(fromDateStr);

                const toDate = new Date(toDateStr);

                const filteredTransactions = transactions.filter(t => {

                    const transactionDate = new Date(t.date);

                    return transactionDate >= fromDate && transactionDate <= toDate;

                });

                let totalReportValue = 0;

                materials.forEach(m => {

                    const stock = calculateStock(m.id, filteredTransactions);

                    const value = stock \* m.price;

                    totalReportValue += value;

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

                    row.className = 'border-b border-gray-200 hover:bg-gray-50';

                    row.innerHTML = `

                        <td class="p-3">${m.id}</td>

                        <td class="p-3 font-medium">${m.description}</td>

                        <td class="p-3 text-right">${m.price.toFixed(2)}</td>

                        <td class="p-3 text-right font-bold">${stock}</td>

                        <td class="p-3">${m.unit}</td>

                        <td class="p-3 text-right">${value.toFixed(2)}</td>

                    `;

                    tableBody.appendChild(row);

                });

                // Add a total row at the end

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

                totalRow.className = 'border-b border-gray-200 font-bold bg-gray-100';

                totalRow.innerHTML = `

                    <td class="p-3" colspan="5">รวมมูลค่าทั้งหมดในรายงาน</td>

                    <td class="p-3 text-right">${totalReportValue.toFixed(2)}</td>

                `;

                tableBody.appendChild(totalRow);

            } catch (error) {

                console.error('Error rendering summary report:', error);

                showAlert('ข้อผิดพลาด', 'ไม่สามารถสร้างรายงานได้ กรุณาลองใหม่อีกครั้ง');

            }

        }

        function renderHistoryTable() {

            try {

                const tableBody = document.getElementById('history-table');

                const typeFilter = document.getElementById('filter-type').value;

                const materialFilter = document.getElementById('filter-material').value;

                const monthFilter = document.getElementById('filter-month').value;

                tableBody.innerHTML = '';

                let filteredTransactions = transactions.filter(t => {

                    const typeMatch = typeFilter === 'all' || t.type === typeFilter;

                    const materialMatch = materialFilter === 'all' || t.materialId === materialFilter;

                    const monthMatch = !monthFilter || t.date.startsWith(monthFilter);

                    return typeMatch && materialMatch && monthMatch;

                });

                filteredTransactions.sort((a,b) => new Date(b.date) - new Date(a.date));

                filteredTransactions.forEach(t => {

                    const material = materials.find(m => m.id === t.materialId);

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

                    row.className = 'border-b border-gray-200';

                    row.innerHTML = `

                        <td class="p-3">${new Date(t.date).toLocaleDateString('th-TH')}</td>

                        <td class="p-3">

                            <span class="px-2 py-1 text-xs rounded-full ${t.type === 'receipt' ? 'bg-green-100 text-green-800' : 'bg-red-100 text-red-800'}">

                                ${t.type === 'receipt' ? 'รับ' : 'เบิก'}

                            </span>

                        </td>

                        <td class="p-3 font-medium">${material ? material.description : 'N/A'}</td>

                        <td class="p-3 text-right">${t.quantity}</td>

                        <td class="p-3">${t.project || '-'}</td>

                        <td class="p-3">${t.ref || t.po || t.requester}</td>

                    `;

                    tableBody.appendChild(row);

                });

            } catch (error) {

                 console.error('Error rendering history table:', error);

            }

        }

        function populateSelectOptions() {

            const selects = document.querySelectorAll('select[name="materialId"], #filter-material');

            const options = materials.map(m => `<option value="${m.id}">${m.id} - ${m.description}</option>`).join('');

            selects.forEach(select => {

                const selectedValue = select.value;

                const firstOption = select.querySelector('option');

                select.innerHTML = '';

                if(firstOption) {

                    select.appendChild(firstOption);

                }

                select.innerHTML += options;

                if (selectedValue) {

                    select.value = selectedValue;

                }

            });

        }

        function switchView(viewName) {

            const views = document.querySelectorAll('[id^="view-"]');

            const navs = document.querySelectorAll('[id^="nav-"]');

            views.forEach(v => v.classList.add('hidden'));

            navs.forEach(n => n.classList.remove('active'));

            document.getElementById(`view-${viewName}`).classList.remove('hidden');

            document.getElementById(`nav-${viewName}`).classList.add('active');

            if (viewName === 'dashboard') {

                renderInventoryTable();

                renderInventoryChart();

            } else if (viewName === 'history') {

                renderHistoryTable();

            } else if (viewName === 'summary') {

                renderSummaryReport();

            }

        }

        function sortTable(tableId, columnIndex, isNumeric = false) {

            const table = document.getElementById(tableId);

            const tbody = table.tBodies[0];

            const rows = Array.from(tbody.rows);

            const sortDirection = table.getAttribute(`data-sort-dir-${columnIndex}`) === 'asc' ? 'desc' : 'asc';

            rows.sort((a, b) => {

                const cellA = a.cells[columnIndex].innerText;

                const cellB = b.cells[columnIndex].innerText;

                let valA, valB;

                if (isNumeric) {

                    valA = parseFloat(cellA.replace(/[^0-9.-]+/g,""));

                    valB = parseFloat(cellB.replace(/[^0-9.-]+/g,""));

                } else {

                    valA = cellA.toLowerCase();

                    valB = cellB.toLowerCase();

                }

                if (valA < valB) return sortDirection === 'asc' ? -1 : 1;

                if (valA > valB) return sortDirection === 'asc' ? 1 : -1;

                return 0;

            });

            rows.forEach(row => tbody.appendChild(row));

            table.setAttribute(`data-sort-dir-${columnIndex}`, sortDirection);

        }

        document.addEventListener('DOMContentLoaded', () => {

            populateSelectOptions();

            renderInventoryTable();

            renderInventoryChart();

            // Set default dates for the summary report

            document.getElementById('report-from-date').value = '2025-01-01';

            const endOfYear = new Date(new Date().getFullYear(), 11, 31).toISOString().split('T')[0];

            document.getElementById('report-to-date').value = endOfYear;

            document.getElementById('search-inventory').addEventListener('input', (e) => {

                const searchTerm = e.target.value.toLowerCase();

                const filtered = materials.filter(m => m.description.toLowerCase().includes(searchTerm) || m.id.toLowerCase().includes(searchTerm));

                renderInventoryTable(filtered);

            });

            document.getElementById('receipt-form').addEventListener('submit', (e) => {

                e.preventDefault();

                const formData = new FormData(e.target);

                const materialId = formData.get('materialId');

                if (materialId === '') {

                    showAlert('ข้อผิดพลาด', 'กรุณาเลือกรหัสวัสดุ');

                    return;

                }

                const newTransaction = {

                    type: 'receipt',

                    date: formData.get('date'),

                    company: formData.get('company'),

                    po: formData.get('po'),

                    materialId: materialId,

                    quantity: parseInt(formData.get('quantity')),

                    price: parseFloat(formData.get('price')),

                    project: formData.get('project'),

                    ref: formData.get('ref'),

                };

                transactions.push(newTransaction);

                e.target.reset();

                showAlert('สำเร็จ', 'บันทึกการรับวัสดุเรียบร้อยแล้ว');

                switchView('dashboard');

                updateStockDisplay('receipt', null);

            });

            document.getElementById('request-form').addEventListener('submit', (e) => {

                e.preventDefault();

                const formData = new FormData(e.target);

                const materialId = formData.get('materialId');

                if (materialId === '') {

                    showAlert('ข้อผิดพลาด', 'กรุณาเลือกรหัสวัสดุ');

                    return;

                }

                const quantity = parseInt(formData.get('quantity'));

                const currentStock = calculateStock(materialId);

                if (quantity > currentStock) {

                    showAlert('ข้อผิดพลาด', `ไม่สามารถเบิกได้ จำนวนวัสดุไม่เพียงพอ (คงเหลือ: ${currentStock})`);

                    return;

                }

                const newTransaction = {

                    type: 'request',

                    date: formData.get('date'),

                    requester: formData.get('requester'),

                    materialId: materialId,

                    quantity: quantity,

                    project: formData.get('project'),

                    ref: formData.get('ref'),

                };

                transactions.push(newTransaction);

                e.target.reset();

                showAlert('สำเร็จ', 'บันทึกการเบิกวัสดุเรียบร้อยแล้ว');

                switchView('dashboard');

                updateStockDisplay('request', null);

            });

            document.getElementById('filter-type').addEventListener('change', renderHistoryTable);

            document.getElementById('filter-material').addEventListener('change', renderHistoryTable);

            document.getElementById('filter-month').addEventListener('change', renderHistoryTable);

            document.getElementById('generate-report-btn').addEventListener('click', renderSummaryReport);

            document.getElementById('receipt-material-select').addEventListener('change', (e) => {

                updateStockDisplay('receipt', e.target.value);

            });

            document.getElementById('request-material-select').addEventListener('change', (e) => {

                updateStockDisplay('request', e.target.value);

            });

            const todayStrInput = new Date().toISOString().split('T')[0];

            document.querySelectorAll('input[type="date"]').forEach(input => {

                if (input.id !== 'report-from-date' && input.id !== 'report-to-date') {

                    input.value = todayStrInput;

                }

            });

        });

    </script>

</body>

</html>