<!DOCTYPE html>  
<html lang="th">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0">  
 <title>โปรแกรมคำนวณราคาทุเรียนภูเขาไฟ (Admin)</title>  
 <!-- Tailwind CSS -->  
 <script src="https://cdn.tailwindcss.com"></script>  
 <!-- jsPDF and html2canvas -->  
 <script src="https://cdnjs.cloudflare.com/ajax/libs/jspdf/2.5.1/jspdf.umd.min.js"></script>  
 <script src="https://cdnjs.cloudflare.com/ajax/libs/html2canvas/1.4.1/html2canvas.min.js"></script>  
 <!-- Google Fonts -->  
 <link href="https://fonts.googleapis.com/css2?family=Kanit:wght@300;400;500;600;700&display=swap" rel="stylesheet">  
 <!-- Icon Library -->  
 <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">  
 <style>  
 body { font-family: 'Kanit', sans-serif; }  
 .form-input { @apply w-full px-4 py-2 bg-gray-50 border border-gray-300 rounded-lg focus:outline-none focus:ring-2 focus:ring-yellow-500 transition-shadow; }  
 .radio-label { @apply flex items-center p-3 border border-gray-200 rounded-lg cursor-pointer hover:bg-yellow-50 hover:border-yellow-400 transition-all; }  
 .radio-input:checked + .radio-label { @apply bg-yellow-100 border-yellow-500 ring-2 ring-yellow-400; }  
 .btn { @apply w-full font-bold py-3 px-4 rounded-lg focus:outline-none focus:ring-2 focus:ring-offset-2 transition-all duration-300 shadow-md hover:shadow-lg flex items-center justify-center gap-2; }  
 .btn-primary { @apply bg-yellow-500 text-white hover:bg-yellow-600 focus:ring-yellow-500; }  
 .btn-secondary { @apply bg-green-600 text-white hover:bg-green-700 focus:ring-green-500; }  
 .btn-success { @apply bg-blue-600 text-white hover:bg-blue-700 focus:ring-blue-500; }  
 .btn-danger { @apply bg-red-600 text-white hover:bg-red-700 focus:ring-red-500; }  
 #admin-login-modal, #quotation-detail-modal { @apply fixed inset-0 bg-gray-900 bg-opacity-75 flex items-center justify-center z-50 transition-opacity; }  
 </style>  
</head>  
<body class="bg-yellow-50/50">  
  
 <!-- Header -->  
 <header class="bg-white shadow-md p-4 flex justify-between items-center">  
 <h1 class="text-xl font-bold text-gray-800">ทุเรียนภูเขาไฟ</h1>  
 <button id="show-admin-login" class="text-gray-600 hover:text-yellow-600 flex items-center gap-2">  
 <i class="fa-solid fa-database"></i>  
 <span>ระบบหลังบ้าน</span>  
 </button>  
 </header>  
  
 <!-- Main App View -->  
 <div id="app-view">  
 <main class="w-full max-w-4xl mx-auto bg-white p-6 md:p-10 rounded-2xl shadow-lg border border-gray-200 my-8">  
 <div class="text-center mb-8">  
 <h1 class="text-3xl md:text-4xl font-bold text-gray-800">โปรแกรมคำนวณราคาทุเรียน</h1>  
 <p class="text-gray-500 mt-2">กรอกข้อมูลเพื่อคำนวณราคาและสร้างใบเสนอราคา</p>  
 </div>  
 <div class="grid md:grid-cols-2 gap-8">  
 <!-- Form Section -->  
 <div class="space-y-6">  
 <form id="price-calculator-form">  
 <div>  
 <label for="customerName" class="block text-sm font-medium text-gray-700 mb-1">ชื่อลูกค้า</label>  
 <input type="text" id="customerName" class="form-input" required>  
 </div>  
 <div>  
 <label for="productName" class="block text-sm font-medium text-gray-700 mb-1">ชื่อสินค้า</label>  
 <input type="text" id="productName" class="form-input bg-gray-200" value="ทุเรียนภูเขาไฟศรีสะเกษ" readonly>  
 </div>  
 <div>  
 <label for="weight" class="block text-sm font-medium text-gray-700 mb-1">น้ำหนัก (กก.)</label>  
 <input type="number" id="weight" class="form-input" min="0" step="0.01" required>  
 </div>  
 <div>  
 <label class="block text-sm font-medium text-gray-700 mb-2">ราคาขาย (บาท/กก.)</label>  
 <div class="grid grid-cols-2 sm:grid-cols-4 gap-2">  
 <div><input type="radio" id="price120" name="priceOption" value="120" class="hidden radio-input" checked><label for="price120" class="radio-label justify-center">120.-</label></div>  
 <div><input type="radio" id="price130" name="priceOption" value="130" class="hidden radio-input"><label for="price130" class="radio-label justify-center">130.-</label></div>  
 <div><input type="radio" id="price150" name="priceOption" value="150" class="hidden radio-input"><label for="price150" class="radio-label justify-center">150.-</label></div>  
 <div><input type="radio" id="priceCustom" name="priceOption" value="custom" class="hidden radio-input"><label for="priceCustom" class="radio-label justify-center">อื่นๆ</label></div>  
 </div>  
 <input type="number" id="customPrice" class="form-input mt-2 hidden" placeholder="กรอกราคาเอง" min="0">  
 </div>  
 <div>  
 <label for="logoUpload" class="block text-sm font-medium text-gray-700 mb-1">อัปโหลดโลโก้ร้าน</label>  
 <input type="file" id="logoUpload" accept="image/\*" class="w-full text-sm text-gray-500 file:mr-4 file:py-2 file:px-4 file:rounded-lg file:border-0 file:text-sm file:font-semibold file:bg-yellow-100 file:text-yellow-800 hover:file:bg-yellow-200 cursor-pointer">  
 </div>  
 <button type="submit" class="btn btn-primary"><i class="fa-solid fa-calculator"></i>คำนวณราคา</button>  
 </form>  
 </div>  
 <!-- Quotation Result Section -->  
 <div id="quotation-container" class="bg-gray-50 p-6 rounded-lg border border-dashed border-gray-300 hidden">  
 <div id="quotation-content">  
 <!-- Content generated by JS -->  
 </div>  
 <div id="action-buttons" class="grid grid-cols-1 sm:grid-cols-2 gap-3 mt-4">  
 <button id="save-to-db" class="btn btn-success"><i class="fa-solid fa-save"></i>บันทึกใบเสนอราคา</button>  
 <button id="download-pdf" class="btn btn-secondary"><i class="fa-solid fa-file-pdf"></i>ดาวน์โหลด PDF</button>  
 </div>  
 </div>  
 </div>  
 </main>  
 </div>  
  
 <!-- Admin View (Hidden by default) -->  
 <div id="admin-view" class="hidden">  
 <div class="w-full max-w-6xl mx-auto p-4 md:p-8">  
 <div class="flex justify-between items-center mb-6">  
 <h2 class="text-3xl font-bold text-gray-800">ข้อมูลใบเสนอราคาทั้งหมด</h2>  
 <button id="back-to-app" class="text-gray-600 hover:text-yellow-600 flex items-center gap-2">  
 <i class="fa-solid fa-arrow-left"></i>  
 <span>กลับหน้าหลัก</span>  
 </button>  
 </div>  
 <div id="quotation-list" class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-6">  
 <!-- Quotation cards will be injected here -->  
 </div>  
 </div>  
 </div>  
  
 <!-- Admin Login Modal -->  
 <div id="admin-login-modal" class="hidden">  
 <div class="bg-white p-8 rounded-lg shadow-xl text-center">  
 <h3 class="text-xl font-bold mb-4">กรุณาใส่รหัสผ่าน</h3>  
 <input type="password" id="admin-password" class="form-input text-center" placeholder="\*\*\*\*\*\*">  
 <div class="grid grid-cols-2 gap-4 mt-6">  
 <button id="cancel-login" class="btn bg-gray-300 hover:bg-gray-400">ยกเลิก</button>  
 <button id="submit-login" class="btn btn-primary">เข้าสู่ระบบ</button>  
 </div>  
 </div>  
 </div>  
   
 <!-- Quotation Detail Modal -->  
 <div id="quotation-detail-modal" class="hidden opacity-0">  
 <div class="bg-white rounded-lg shadow-xl w-full max-w-2xl max-h-[90vh] overflow-y-auto relative">  
 <button id="close-detail-modal" class="absolute top-4 right-4 text-gray-500 hover:text-red-500 text-2xl z-10">  
 <i class="fa-solid fa-times-circle"></i>  
 </button>  
 <div id="modal-quotation-content" class="p-8">  
 <!-- Full quotation content will be injected here -->  
 </div>  
 </div>  
 </div>  
  
 <!-- Firebase SDKs -->  
 <script type="module">  
 // Import Firebase functions  
 import { initializeApp } from "https://www.gstatic.com/firebasejs/10.12.2/firebase-app.js";  
 import { getFirestore, collection, addDoc, getDocs, onSnapshot, doc, deleteDoc, query, orderBy } from "https://www.gstatic.com/firebasejs/10.12.2/firebase-firestore.js";  
 import { getAuth, signInAnonymously, onAuthStateChanged } from "https://www.gstatic.com/firebasejs/10.12.2/firebase-auth.js";  
  
 // --- Your web app's Firebase configuration (PROVIDED BY YOU) ---  
 // =========================================================================  
 const firebaseConfig = {  
 apiKey: "AIzaSyD3\_szfgr4xdeEBYU4H0sx1LUwoGW-gzuk",  
 authDomain: "durianos1.firebaseapp.com",  
 projectId: "durianos1",  
 storageBucket: "durianos1.firebasestorage.app",  
 messagingSenderId: "44995004576",  
 appId: "1:44995004576:web:a36e4498c6ff66d89e61c5",  
 measurementId: "G-F391822XE9"  
 };  
 // =========================================================================  
   
 // --- Initialize Firebase ---  
 let app, db, auth;  
 try {  
 app = initializeApp(firebaseConfig);  
 db = getFirestore(app);  
 auth = getAuth(app);  
 } catch (e) {  
 console.error("Firebase initialization failed. Please check your firebaseConfig.", e);  
 alert("การตั้งค่า Firebase ไม่ถูกต้อง! กรุณาตรวจสอบค่า firebaseConfig ในโค้ด");  
 return;  
 }  
  
 // --- Authenticate Anonymously ---  
 onAuthStateChanged(auth, (user) => {  
 if (user) {  
 console.log("User is signed in anonymously.", user.uid);  
 // Now that we are authenticated, we can fetch data.  
 loadQuotations();  
 } else {  
 console.log("User is signed out.");  
 signInAnonymously(auth).catch((error) => {  
 console.error("Anonymous sign-in failed:", error);  
 });  
 }  
 });  
  
 // --- Global State ---  
 let currentQuotationData = null;  
 let logoDataUrl = 'https://i.imgur.com/832P5W0.png'; // Default logo  
  
 // --- DOM Elements ---  
 const priceForm = document.getElementById('price-calculator-form');  
 const customPriceInput = document.getElementById('customPrice');  
 const logoUploadInput = document.getElementById('logoUpload');  
 const quotationContainer = document.getElementById('quotation-container');  
 const quotationContentDiv = document.getElementById('quotation-content');  
   
 const saveToDbBtn = document.getElementById('save-to-db');  
 const downloadPdfBtn = document.getElementById('download-pdf');  
   
 const appView = document.getElementById('app-view');  
 const adminView = document.getElementById('admin-view');  
 const quotationListDiv = document.getElementById('quotation-list');  
  
 const showAdminLoginBtn = document.getElementById('show-admin-login');  
 const adminLoginModal = document.getElementById('admin-login-modal');  
 const cancelLoginBtn = document.getElementById('cancel-login');  
 const submitLoginBtn = document.getElementById('submit-login');  
 const adminPasswordField = document.getElementById('admin-password');  
 const backToAppBtn = document.getElementById('back-to-app');  
  
 const detailModal = document.getElementById('quotation-detail-modal');  
 const detailContentDiv = document.getElementById('modal-quotation-content');  
 const closeDetailModalBtn = document.getElementById('close-detail-modal');  
  
 // --- Functions ---  
 const generateQuotationHTML = (data) => {  
 const currencyFormatter = new Intl.NumberFormat('th-TH', { style: 'currency', currency: 'THB' });  
 const numberFormatter = new Intl.NumberFormat('th-TH', { minimumFractionDigits: 2, maximumFractionDigits: 2 });  
 const quoteDate = data.quoteDate ? new Date(data.quoteDate) : new Date();  
  
 return `  
 <div class="flex justify-between items-start pb-4 border-b-2 border-gray-200 mb-4">  
 <div>  
 <h2 class="text-2xl font-bold text-gray-800">ใบเสนอราคา</h2>  
 <p class="text-sm text-gray-500">${quoteDate.toLocaleDateString('th-TH', { year: 'numeric', month: 'long', day: 'numeric' })}</p>  
 </div>  
 <img src="${data.logoUrl || logoDataUrl}" alt="Logo" class="max-h-16 object-contain">  
 </div>  
 <div class="mb-6">  
 <p class="text-sm text-gray-500">เรียนคุณ:</p>  
 <p class="font-semibold text-gray-800 text-lg">${data.customerName}</p>  
 </div>  
 <table class="w-full text-left">  
 <thead class="bg-gray-100">  
 <tr>  
 <th class="p-3 text-sm font-semibold text-gray-600">รายการ</th>  
 <th class="p-3 text-sm font-semibold text-gray-600 text-center">น้ำหนัก</th>  
 <th class="p-3 text-sm font-semibold text-gray-600 text-center">ราคา/หน่วย</th>  
 <th class="p-3 text-sm font-semibold text-gray-600 text-right">ราคารวม</th>  
 </tr>  
 </thead>  
 <tbody>  
 <tr class="border-b">  
 <td class="p-3">${data.productName}</td>  
 <td class="p-3 text-center">${numberFormatter.format(data.weight)}</td>  
 <td class="p-3 text-center">${currencyFormatter.format(data.unitPrice)}</td>  
 <td class="p-3 text-right">${currencyFormatter.format(data.totalPrice)}</td>  
 </tr>  
 </tbody>  
 </table>  
 <div class="mt-6 flex justify-end">  
 <div class="w-full max-w-xs text-right">  
 <div class="flex justify-between text-2xl mt-2 pt-2 border-t-2">  
 <span class="font-bold text-gray-800">ยอดรวมสุทธิ</span>  
 <span class="font-bold text-yellow-600">${currencyFormatter.format(data.totalPrice)}</span>  
 </div>  
 </div>  
 </div>  
 `;  
 };  
   
 const loadQuotations = () => {  
 const quotationsCol = collection(db, 'quotations');  
 // Order by date descending to show newest first  
 const q = query(quotationsCol, orderBy("quoteDate", "desc"));  
  
 onSnapshot(q, (snapshot) => {  
 quotationListDiv.innerHTML = ''; // Clear current list  
 if (snapshot.empty) {  
 quotationListDiv.innerHTML = `<p class="text-gray-500 col-span-full text-center">ยังไม่มีข้อมูลใบเสนอราคาที่บันทึกไว้</p>`;  
 return;  
 }  
 let allDocs = [];  
 snapshot.forEach(doc => {  
 allDocs.push({ id: doc.id, ...doc.data() });  
 });  
  
 // Sort in memory instead of relying on Firestore's index  
 allDocs.sort((a, b) => new Date(b.quoteDate) - new Date(a.quoteDate));  
  
 allDocs.forEach(data => {  
 const quoteCard = document.createElement('div');  
 quoteCard.className = 'bg-white p-5 rounded-lg shadow border transition hover:shadow-md hover:border-yellow-400';  
 quoteCard.innerHTML = `  
 <div class="flex justify-between items-start">  
 <div class="flex-1">  
 <p class="font-bold text-lg text-gray-800">${data.customerName}</p>  
 <p class="text-sm text-gray-500">${new Date(data.quoteDate).toLocaleDateString('th-TH')}</p>  
 </div>  
 <img src="${data.logoUrl || logoDataUrl}" class="h-8 ml-2 object-contain">  
 </div>  
 <p class="text-2xl font-semibold text-yellow-600 my-3 text-right">${new Intl.NumberFormat('th-TH', { style: 'currency', currency: 'THB' }).format(data.totalPrice)}</p>  
 <div class="grid grid-cols-2 gap-2 mt-4">  
 <button class="view-btn btn btn-sm bg-gray-200 hover:bg-gray-300" data-id="${data.id}"><i class="fa-solid fa-eye"></i>ดู</button>  
 <button class="delete-btn btn btn-sm bg-red-100 text-red-700 hover:bg-red-200" data-id="${data.id}"><i class="fa-solid fa-trash"></i>ลบ</button>  
 </div>  
 `;  
 quotationListDiv.appendChild(quoteCard);  
 });  
 });  
 };  
  
 // --- Event Listeners ---  
 document.querySelectorAll('input[name="priceOption"]').forEach(radio => {  
 radio.addEventListener('change', () => {  
 customPriceInput.classList.toggle('hidden', radio.value !== 'custom');  
 customPriceInput.required = (radio.value === 'custom');  
 });  
 });  
  
 logoUploadInput.addEventListener('change', (event) => {  
 const file = event.target.files[0];  
 if (file) {  
 const reader = new FileReader();  
 reader.onload = (e) => { logoDataUrl = e.target.result; };  
 reader.readAsDataURL(file);  
 }  
 });  
  
 priceForm.addEventListener('submit', (e) => {  
 e.preventDefault();  
 const selectedPriceOption = document.querySelector('input[name="priceOption"]:checked').value;  
 const price = (selectedPriceOption === 'custom') ? parseFloat(customPriceInput.value) : parseFloat(selectedPriceOption);  
 const weight = parseFloat(document.getElementById('weight').value);  
  
 currentQuotationData = {  
 customerName: document.getElementById('customerName').value,  
 productName: document.getElementById('productName').value,  
 weight: weight,  
 unitPrice: price,  
 totalPrice: weight \* price,  
 quoteDate: new Date().toISOString(),  
 logoUrl: logoDataUrl,  
 };  
  
 quotationContentDiv.innerHTML = generateQuotationHTML(currentQuotationData);  
 quotationContainer.classList.remove('hidden');  
 });  
  
 saveToDbBtn.addEventListener('click', async () => {  
 if (!currentQuotationData) return;  
 try {  
 const docRef = await addDoc(collection(db, "quotations"), currentQuotationData);  
 console.log("Document written with ID: ", docRef.id);  
 alert('บันทึกใบเสนอราคาสำเร็จ!');  
 // Reset form  
 priceForm.reset();  
 quotationContainer.classList.add('hidden');  
 currentQuotationData = null;  
 } catch (error) {  
 console.error("Error adding document: ", error);  
 alert('เกิดข้อผิดพลาดในการบันทึกข้อมูล');  
 }  
 });  
  
 downloadPdfBtn.addEventListener('click', async () => {  
 const { jsPDF } = window.jspdf;  
 const tempDiv = document.createElement('div');  
 tempDiv.innerHTML = generateQuotationHTML(currentQuotationData);  
 tempDiv.style.width = '800px'; // Set a fixed width for consistent PDF rendering  
 document.body.appendChild(tempDiv);  
   
 const canvas = await html2canvas(tempDiv, { scale: 2, useCORS: true });  
 document.body.removeChild(tempDiv);  
   
 const imgData = canvas.toDataURL('image/png');  
 const pdf = new jsPDF({ orientation: 'p', unit: 'mm', format: 'a4' });  
 const pdfWidth = pdf.internal.pageSize.getWidth();  
 const pdfHeight = (canvas.height \* pdfWidth) / canvas.width;  
 pdf.addImage(imgData, 'PNG', 0, 0, pdfWidth, pdfHeight);  
   
 const filename = `ใบเสนอราคา\_${currentQuotationData.customerName.replace(/\s/g, '\_')}.pdf`;  
 pdf.save(filename);  
 });  
  
 showAdminLoginBtn.addEventListener('click', () => adminLoginModal.classList.remove('hidden'));  
 cancelLoginBtn.addEventListener('click', () => adminLoginModal.classList.add('hidden'));  
 backToAppBtn.addEventListener('click', () => {  
 adminView.classList.add('hidden');  
 appView.classList.remove('hidden');  
 });  
  
 submitLoginBtn.addEventListener('click', () => {  
 if (adminPasswordField.value === '14506') {  
 adminLoginModal.classList.add('hidden');  
 appView.classList.add('hidden');  
 adminView.classList.remove('hidden');  
 adminPasswordField.value = '';  
 } else {  
 alert('รหัสผ่านไม่ถูกต้อง!');  
 }  
 });  
  
 quotationListDiv.addEventListener('click', (e) => {  
 const target = e.target.closest('button');  
 if (!target) return;  
  
 const id = target.dataset.id;  
 if (target.classList.contains('delete-btn')) {  
 if (confirm('คุณแน่ใจหรือไม่ว่าต้องการลบใบเสนอราคานี้?')) {  
 deleteDoc(doc(db, 'quotations', id))  
 .then(() => console.log("Document successfully deleted!"))  
 .catch((error) => console.error("Error removing document: ", error));  
 }  
 } else if (target.classList.contains('view-btn')) {  
 getDocs(collection(db, 'quotations')).then(snapshot => {  
 snapshot.forEach(docSnapshot => {  
 if (docSnapshot.id === id) {  
 detailContentDiv.innerHTML = generateQuotationHTML(docSnapshot.data());  
 detailModal.classList.remove('hidden', 'opacity-0');  
 }  
 })  
 });  
 }  
 });  
   
 closeDetailModalBtn.addEventListener('click', () => {  
 detailModal.classList.add('hidden', 'opacity-0');  
 });  
   
 </script>  
</body>  
</html>