<!DOCTYPE html>

<html lang="en" class="bg-gray-100 dark:bg-gray-900">

<head>

<meta charset="UTF-8">

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

<title>Bible App</title>

<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=Inter:wght@400;500;600;700&display=swap" rel="stylesheet">

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

<style>

body {

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

transition: background-color 0.3s ease;

}

.container {

max-width: 800px;

}

.text-content {

white-space: pre-wrap;

}

.dark-mode-toggle {

cursor: pointer;

padding: 8px 12px;

border-radius: 9999px;

background-color: #e5e7eb;

color: #1f2937;

border: none;

outline: none;

transition: background-color 0.3s ease, transform 0.2s ease;

}

.dark-mode-toggle:hover {

transform: scale(1.05);

}

.dark .dark-mode-toggle {

background-color: #374151;

color: #d1d5db;

}

select, input, button {

transition: all 0.2s ease-in-out;

}

.main-content {

max-height: calc(100vh - 180px); /\* Adjust height for controls and header \*/

overflow-y: auto;

}

</style>

</head>

<body class="dark:bg-gray-900 dark:text-gray-100">

<div class="container mx-auto p-4 flex flex-col min-h-screen">

<!-- Header -->

<header class="flex flex-col sm:flex-row items-center justify-between gap-4 mb-6">

<h1 class="text-3xl font-bold text-gray-800 dark:text-white">Bible Reader</h1>

<button id="darkModeToggle" class="dark-mode-toggle flex items-center gap-2">

<span id="darkModeIcon">☀️</span>

<span id="darkModeText">Light Mode</span>

</button>

</header>

<!-- Controls Section -->

<div class="bg-white dark:bg-gray-800 p-4 sm:p-6 rounded-lg shadow-lg mb-6 flex flex-col sm:flex-row items-center justify-center gap-4">

<select id="bookSelect" class="w-full sm:w-1/3 p-2 rounded-md border border-gray-300 dark:border-gray-600 bg-gray-50 dark:bg-gray-700 dark:text-white">

<option value="">Select a Book</option>

</select>

<select id="chapterSelect" class="w-full sm:w-1/3 p-2 rounded-md border border-gray-300 dark:border-gray-600 bg-gray-50 dark:bg-gray-700 dark:text-white" disabled>

<option value="">Select a Chapter</option>

</select>

<button id="loadChapterButton" class="w-full sm:w-1/3 bg-blue-500 text-white font-semibold py-2 px-4 rounded-md hover:bg-blue-600 focus:outline-none focus:ring-2 focus:ring-blue-500 focus:ring-opacity-50 transition-colors duration-200" disabled>

Read

</button>

</div>

<!-- Bible Text Display -->

<main class="main-content flex-grow bg-white dark:bg-gray-800 p-6 rounded-lg shadow-lg">

<div id="bibleText" class="text-lg leading-relaxed text-gray-700 dark:text-gray-300">

<p class="text-center text-gray-500 dark:text-gray-400">Please select a book and chapter to start reading.</p>

</div>

<div id="loadingIndicator" class="hidden text-center mt-4">

<div class="animate-spin rounded-full h-8 w-8 border-b-2 border-gray-900 mx-auto"></div>

<p class="mt-2">Loading...</p>

</div>

</main>

</div>

<script>

const API\_BASE\_URL = 'https://bible-api.com/';

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

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

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

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

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

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

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

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

// Dark mode setup

const isDarkMode = localStorage.getItem('darkMode') === 'true';

if (isDarkMode) {

document.documentElement.classList.add('dark');

darkModeIcon.textContent = '🌙';

darkModeText.textContent = 'Dark Mode';

} else {

document.documentElement.classList.remove('dark');

darkModeIcon.textContent = '☀️';

darkModeText.textContent = 'Light Mode';

}

darkModeToggle.addEventListener('click', () => {

const isDark = document.documentElement.classList.toggle('dark');

localStorage.setItem('darkMode', isDark);

darkModeIcon.textContent = isDark ? '🌙' : '☀️';

darkModeText.textContent = isDark ? 'Dark Mode' : 'Light Mode';

});

// Mapping of book names to their number of chapters

const bibleBooks = {

"Genesis": 50, "Exodus": 40, "Leviticus": 27, "Numbers": 36, "Deuteronomy": 34,

"Joshua": 24, "Judges": 21, "Ruth": 4, "1 Samuel": 31, "2 Samuel": 24,

"1 Kings": 22, "2 Kings": 25, "1 Chronicles": 29, "2 Chronicles": 36, "Ezra": 10,

"Nehemiah": 13, "Esther": 10, "Job": 42, "Psalms": 150, "Proverbs": 31,

"Ecclesiastes": 12, "Song of Solomon": 8, "Isaiah": 66, "Jeremiah": 52, "Lamentations": 5,

"Ezekiel": 48, "Daniel": 12, "Hosea": 14, "Joel": 3, "Amos": 9,

"Obadiah": 1, "Jonah": 4, "Micah": 7, "Nahum": 3, "Habakkuk": 3,

"Zephaniah": 3, "Haggai": 2, "Zechariah": 14, "Malachi": 4, "Matthew": 28,

"Mark": 16, "Luke": 24, "John": 21, "Acts": 28, "Romans": 16,

"1 Corinthians": 16, "2 Corinthians": 13, "Galatians": 6, "Ephesians": 6, "Philippians": 4,

"Colossians": 4, "1 Thessalonians": 5, "2 Thessalonians": 3, "1 Timothy": 6, "2 Timothy": 4,

"Titus": 3, "Philemon": 1, "Hebrews": 13, "James": 5, "1 Peter": 5,

"2 Peter": 3, "1 John": 5, "2 John": 1, "3 John": 1, "Jude": 1,

"Revelation": 22

};

// Populate book dropdown

function populateBooks() {

for (const book in bibleBooks) {

const option = document.createElement('option');

option.value = book;

option.textContent = book;

bookSelect.appendChild(option);

}

}

// Populate chapter dropdown based on selected book

bookSelect.addEventListener('change', () => {

const selectedBook = bookSelect.value;

chapterSelect.innerHTML = '<option value="">Select a Chapter</option>';

chapterSelect.disabled = true;

loadChapterButton.disabled = true;

if (selectedBook) {

const numChapters = bibleBooks[selectedBook];

for (let i = 1; i <= numChapters; i++) {

const option = document.createElement('option');

option.value = i;

option.textContent = `Chapter ${i}`;

chapterSelect.appendChild(option);

}

chapterSelect.disabled = false;

}

});

// Enable button when chapter is selected

chapterSelect.addEventListener('change', () => {

loadChapterButton.disabled = !chapterSelect.value;

});

// Fetch and display the selected chapter

loadChapterButton.addEventListener('click', async () => {

const book = bookSelect.value;

const chapter = chapterSelect.value;

if (!book || !chapter) return;

// Show loading indicator

loadingIndicator.classList.remove('hidden');

bibleText.classList.add('hidden');

try {

const response = await fetch(`${API\_BASE\_URL}${book} ${chapter}?translation=kjv`);

if (!response.ok) {

throw new Error('Failed to fetch the chapter.');

}

const data = await response.json();

if (data.error) {

bibleText.innerHTML = `<p class="text-red-500">${data.error}</p>`;

} else if (data.verses) {

const chapterText = data.verses.map(verse =>

`<span class="font-semibold text-blue-500 dark:text-blue-400">${verse.verse}.</span> ${verse.text}`

).join(' ');

bibleText.innerHTML = chapterText;

} else {

bibleText.innerHTML = '<p class="text-gray-500 dark:text-gray-400">No verses found for this chapter.</p>';

}

} catch (error) {

console.error('Error fetching Bible data:', error);

bibleText.innerHTML = `<p class="text-red-500">Could not load the chapter. Please check your connection or try again.</p>`;

} finally {

// Hide loading indicator

loadingIndicator.classList.add('hidden');

bibleText.classList.remove('hidden');

}

});

// Initialize the app

populateBooks();

</script>

</body>

</html>