<!DOCTYPE html>

<html lang="vi">

<head>

<meta charset="UTF-8" />

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

<title>Trang Web Âm Nhạc</title>

<link

href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"

rel="stylesheet"

/>

<link

rel="stylesheet"

href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.6.0/css/all.min.css"

/>

<link rel="stylesheet" href="style.css" />

</head>

<body>

<header class="fixed-top py-1">

<nav class="container navbar navbar-expand-lg">

<div class="container-fluid">

<a class="navbar-brand me-5" href="../index.html">

<img src="../img/Logo.svg" alt="" />

</a>

<button

class="navbar-toggler"

type="button"

data-bs-toggle="offcanvas"

data-bs-target="#offcanvasMenu"

aria-controls="offcanvasMenu"

>

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse">

<ul

class="navbar-nav me-5 mb-2 mb-lg-0 d-flex justify-content-between w-100"

>

<!-- OFFICIAL AUDIO -->

<audio id="audioPlayer" preload="auto">

<source src="path/to/your-audio-file.mp3" type="audio/mpeg" />

</audio>

<li class="nav-item icon">

<a class="nav-link active" href="#" onclick="toggleAudio()">

<i class="fa-solid fa-play" id="playIcon"></i>

</a>

</li>

<!-- COVER AUDIO -->

<li class="nav-item icon">

<a class="nav-link" href="#">

<i class="fa-solid fa-microphone-lines"></i>

</a>

</li>

<li class="nav-item icon">

<a class="nav-link" href="#">

<i class="fa-solid fa-guitar"></i>

</a>

</li>

<li class="nav-item icon">

<a class="nav-link" href="#">

<i class="fa-solid fa-drum"></i>

</a>

</li>

<li class="nav-item icon">

<a class="nav-link" href="#">

<i class="fa-solid fa-file-audio"></i>

</a>

</li>

<li class="nav-item icon">

<a class="nav-link" href="#">

<i class="fa-solid fa-keyboard"></i>

</a>

</li>

</ul>

<!-- Thanh tìm kiếm -->

<div class="search-container w-25 mx-auto">

<i

class="fa-solid fa-magnifying-glass text-primary me-2 fs-6 ms-1"

></i>

<input

type="text"

id="searchInput"

class="form-control border-0 bg-primary bg-opacity-75 rounded-pill py-1 px-2 fs-6"

placeholder="Search for songs..."

onkeyup="searchSong(event)"

/>

<ul id="suggestionsList" class="suggestions p-2"></ul>

</div>

</div>

</div>

</nav>

</header>

<main class="py-5">

<div class="container mt-4">

<h2 class="text-center opacity-0">

Danh sách bài hát (<span id="totalSongs">0</span>)

</h2>

<div class="row">

<div class="col-lg-3 border-end d-none d-lg-block">

<h4>List of Songs (<span id="totalSongsDisplay">0</span>)</h4>

<ul id="artistSongList" class="list-group"></ul>

</div>

<div class="col-lg-6 border border-dark mb-3">

<h2 class="text-danger text-center" id="songTitle"></h2>

<h4 class="text-end" id="songArtist"></h4>

<div id="lyricsContainer" class="lyrics-container"></div>

</div>

<div class="col-lg-3 py-4">

<p><strong>Number:</strong> <span id="songNumber"></span></p>

<p><strong>Rating:</strong> <span id="songRating"></span></p>

<p><strong>Year:</strong> <span id="songYear"></span></p>

<p><strong>Link:</strong> <span id="songLink"></span></p>

<p>

<strong>Description:</strong> <span id="songDescription"></span>

</p>

</div>

</div>

</div>

</main>

<!-- Offcanvas menu -->

<div

class="offcanvas offcanvas-end w-75"

tabindex="-1"

id="offcanvasMenu"

aria-labelledby="offcanvasMenuLabel"

>

<div class="offcanvas-header">

<h4>List of Songs</h4>

<button

type="button"

class="btn-close text-reset"

data-bs-dismiss="offcanvas"

aria-label="Close"

></button>

</div>

<div class="offcanvas-body">

<ul id="offcanvasArtistSongList" class="list-group"></ul>

</div>

</div>

<!-- Các biểu tượng cố định ở dưới cùng trên điện thoại -->

<div class="icon-bar d-lg-none bg-primary px-5 py-2">

<!-- Mobile Audio Player -->

<audio id="mobileAudioPlayer" preload="auto">

<source src="path/to/your-audio-file.mp3" type="audio/mpeg" />

</audio>

<a href="#" class="icon-phone" onclick="toggleAudioMobile()">

<i class="fa-solid fa-play" id="playIconMobile"></i>

</a>

<a href="#" class="icon-phone">

<i class="fa-solid fa-microphone-lines"></i>

</a>

<a href="#" class="icon-phone">

<i class="fa-solid fa-guitar"></i>

</a>

<a href="#" class="icon-phone">

<i class="fa-solid fa-drum"></i>

</a>

<a href="#" class="icon-phone">

<i class="fa-solid fa-file-audio"></i>

</a>

<a href="#" class="icon-phone">

<i class="fa-solid fa-keyboard"></i>

</a>

</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>

<!-- Custom JS -->

<script>

let songs = [];

let currentIndex = 0;

let selectedIndex = -1; // Variable to track selected suggestion

let audio = document.getElementById("audioPlayer");

let playIcon = document.getElementById("playIcon");

let mobileAudio = document.getElementById("mobileAudioPlayer");

let playIconMobile = document.getElementById("playIconMobile");

async function loadSongs() {

try {

const response = await fetch("data.json");

if (!response.ok) throw new Error("Không thể tải dữ liệu");

songs = await response.json();

document.getElementById("totalSongs").textContent = songs.length;

document.getElementById("totalSongsDisplay").textContent =

songs.length;

displayArtistsAndSongs();

displaySong();

} catch (error) {

console.error("Lỗi tải bài hát:", error);

}

}

function displayArtistsAndSongs() {

let artists = [...new Set(songs.map((song) => song.artist))];

const renderArtistSongList = (elementId, hideSongs = false) => {

let artistSongList = document.getElementById(elementId);

artistSongList.innerHTML = artists

.map((artist) => {

let artistSongs = songs.filter((song) => song.artist === artist);

let songItems = artistSongs

.map(

(song) =>

`<li class='list-group-item ps-4' onclick="displaySongByTitle('${song.title}')">${song.title}</li>`

)

.join("");

return `

<li class='list-group-item fw-bold d-flex justify-content-between' onclick="toggleSongs('${artist}')">

${artist} <span class='badge bg-primary'>${artistSongs.length}</span>

</li>

<ul class='list-unstyled song-list ${

hideSongs ? "" : ""

}' id='songs-${artist}'>

${songItems}

</ul>

`;

})

.join("");

};

renderArtistSongList("artistSongList", true); // Danh sách chính

renderArtistSongList("offcanvasArtistSongList"); // Offcanvas

}

function toggleSongs(artist) {

document.querySelectorAll(".song-list").forEach((list) => {

list.id === `songs-${artist}`

? list.classList.toggle("show") // Mở/đóng danh sách được chọn

: list.classList.remove("show"); // Thu gọn các danh sách khác

});

}

async function displaySong() {

if (songs.length === 0) return;

const song = songs[currentIndex];

document.getElementById("songTitle").textContent = song.title;

document.getElementById("songArtist").textContent = song.artist;

document.getElementById("songNumber").textContent = song.number;

document.getElementById("songRating").textContent = song.rating;

document.getElementById("songYear").textContent = song.year;

const songLink = document.getElementById("songLink");

songLink.innerHTML = `<a href="${song.link}" target="\_blank">${song.link}</a>`;

document.getElementById("songDescription").textContent =

song.description;

document.getElementById("audioPlayer").src = song.official;

document.getElementById("mobileAudioPlayer").src = song.official;

// Fetch lời bài hát

try {

const lyricsResponse = await fetch(song.lyrics);

if (!lyricsResponse.ok) throw new Error("Không thể tải lời bài hát");

const lyrics = await lyricsResponse.text();

document.getElementById("songLyrics").innerHTML = lyrics;

} catch (error) {

console.error("Lỗi tải lời bài hát:", error);

document.getElementById("songLyrics").textContent =

"Dữ liệu bài hát chưa được cập nhật.";

}

}

function searchSong(event) {

let input = event.target.value.toLowerCase();

let suggestionsList = document.getElementById("suggestionsList");

suggestionsList.innerHTML = "";

if (input) {

let filteredSongs = songs.filter((song) =>

song.title.toLowerCase().includes(input)

);

filteredSongs.forEach((song) => {

let listItem = document.createElement("li");

listItem.textContent = song.title;

listItem.onclick = () => displaySongByTitle(song.title);

suggestionsList.appendChild(listItem);

});

suggestionsList.style.display = "block";

} else {

suggestionsList.style.display = "none";

}

}

function handleKeyPress(event) {

let suggestionsList = document.getElementById("suggestionsList");

let items = suggestionsList.getElementsByTagName("li");

if (event.key === "ArrowDown") {

selectedIndex = Math.min(items.length - 1, selectedIndex + 1);

} else if (event.key === "ArrowUp") {

selectedIndex = Math.max(0, selectedIndex - 1);

} else if (event.key === "Enter" && selectedIndex !== -1) {

items[selectedIndex].click();

}

for (let i = 0; i < items.length; i++) {

items[i].classList.toggle("selected", i === selectedIndex);

}

}

document

.getElementById("searchInput")

.addEventListener("keydown", handleKeyPress);

loadSongs();

// Chú ý

function syncIcons(isPlaying) {

if (isPlaying) {

playIcon.classList.remove("fa-play");

playIcon.classList.add("fa-pause");

playIconMobile.classList.remove("fa-play");

playIconMobile.classList.add("fa-pause");

} else {

playIcon.classList.remove("fa-pause");

playIcon.classList.add("fa-play");

playIconMobile.classList.remove("fa-pause");

playIconMobile.classList.add("fa-play");

}

}

// Chú ý

// Cập nhật lại hàm toggleAudio để hoạt động đúng cho tất cả các bài hát

function toggleAudio() {

if (!audio || !playIcon) return;

// Nếu audio đang bị tạm dừng, phát lại bài hát

if (audio.paused) {

audio.play().catch((err) => console.error("Lỗi phát nhạc:", err));

if (mobileAudio) mobileAudio.pause(); // Dừng nhạc trên điện thoại nếu đang phát

// Cập nhật icon play/pause cho desktop

playIcon.classList.remove("fa-play");

playIcon.classList.add("fa-pause");

// Đảm bảo icon trên điện thoại không bị thay đổi khi nhấn play trên desktop

if (playIconMobile) {

playIconMobile.classList.remove("fa-pause");

playIconMobile.classList.add("fa-play");

}

} else {

audio.pause(); // Tạm dừng bài hát trên desktop

playIcon.classList.remove("fa-pause");

playIcon.classList.add("fa-play");

// Đảm bảo icon trên điện thoại được cập nhật khi nhấn pause trên desktop

if (playIconMobile) {

playIconMobile.classList.remove("fa-pause");

playIconMobile.classList.add("fa-play");

}

}

}

// Chú ý

// Cho điện thoại

function toggleAudioMobile() {

if (!mobileAudio || !playIconMobile) return;

// Nếu audio đang bị tạm dừng, phát lại bài hát trên mobile

if (mobileAudio.paused) {

mobileAudio

.play()

.catch((err) => console.error("Lỗi phát nhạc mobile:", err));

if (audio) audio.pause(); // Dừng nhạc trên desktop nếu đang phát

// Cập nhật icon play/pause cho mobile

playIconMobile.classList.remove("fa-play");

playIconMobile.classList.add("fa-pause");

// Đảm bảo icon trên desktop không bị thay đổi khi nhấn play trên mobile

if (playIcon) {

playIcon.classList.remove("fa-pause");

playIcon.classList.add("fa-play");

}

} else {

mobileAudio.pause(); // Tạm dừng bài hát trên mobile

playIconMobile.classList.remove("fa-pause");

playIconMobile.classList.add("fa-play");

// Đảm bảo icon trên desktop được cập nhật khi nhấn pause trên mobile

if (playIcon) {

playIcon.classList.remove("fa-pause");

playIcon.classList.add("fa-play");

}

}

}

// Chú ý

// Khi phát bài mới, đảm bảo icon cập nhật đúng

function playSong(songUrl) {

if (!songUrl) return;

// Dừng mọi audio trước khi phát bài hát mới

audio.pause();

mobileAudio.pause();

audio.currentTime = 0;

mobileAudio.currentTime = 0;

// Xác định audio cần phát dựa trên kích thước màn hình

if (window.innerWidth <= 768) {

mobileAudio.src = songUrl;

mobileAudio

.play()

.catch((err) => console.error("Lỗi phát nhạc mobile:", err));

audio.src = ""; // Xóa nguồn của audio desktop để chắc chắn nó không phát

} else {

audio.src = songUrl;

audio.play().catch((err) => console.error("Lỗi phát nhạc:", err));

mobileAudio.src = ""; // Xóa nguồn của audio mobile

}

updatePlayIcons(true); // Cập nhật icon

}

// Chú ý

function updatePlayIcons(isPlaying) {

const action = isPlaying ? "add" : "remove";

playIcon.classList[action]("fa-pause");

playIcon.classList.toggle("fa-play", !isPlaying);

playIconMobile.classList[action]("fa-pause");

playIconMobile.classList.toggle("fa-play", !isPlaying);

}

// Chú ý

function handleSongEnd() {

if (window.innerWidth <= 768) {

if (!mobileAudio.paused) playRandomSong();

} else {

if (!audio.paused) playRandomSong();

}

}

audio.addEventListener("ended", handleSongEnd);

mobileAudio.addEventListener("ended", handleSongEnd);

// Cập nhật biểu tượng play/pause khi bài hát kết thúc

audio.addEventListener("ended", () => {

playIcon.classList.remove("fa-pause");

playIcon.classList.add("fa-play");

});

mobileAudio.addEventListener("ended", () => {

playIconMobile.classList.remove("fa-pause");

playIconMobile.classList.add("fa-play");

});

audio.addEventListener("ended", handleSongEnd);

mobileAudio.addEventListener("ended", handleSongEnd);

// Sự kiện khi bài hát kết thúc -> cập nhật icon

audio.addEventListener("ended", () => {

playIcon.classList.remove("fa-pause");

playIcon.classList.add("fa-play");

});

mobileAudio.addEventListener("ended", () => {

playIconMobile.classList.remove("fa-pause");

playIconMobile.classList.add("fa-play");

});

// Chú ý

// Chuyển bài khi click vào tiêu đề

function displaySongByTitle(title) {

let song = songs.find((song) => song.title === title);

if (!song) return;

// Dừng tất cả luồng âm thanh trước khi phát bài mới

audio.pause();

mobileAudio.pause();

currentIndex = songs.indexOf(song);

displaySong();

playSong(song.official);

}

// Chú ý

// Hàm phát bài ngẫu nhiên

function playRandomSong() {

if (songs.length === 0) return;

let randomIndex;

do {

randomIndex = Math.floor(Math.random() \* songs.length);

} while (randomIndex === currentIndex);

currentIndex = randomIndex;

displaySong();

if (window.innerWidth <= 768) {

mobileAudio.src = songs[currentIndex].official;

mobileAudio.currentTime = 0;

mobileAudio

.play()

.catch((err) => console.error("Lỗi phát nhạc mobile:", err));

audio.src = "";

} else {

audio.src = songs[currentIndex].official;

audio.currentTime = 0;

audio.play().catch((err) => console.error("Lỗi phát nhạc:", err));

mobileAudio.src = "";

}

updatePlayIcons(true);

}

// Sự kiện khi bài hát kết thúc -> phát bài ngẫu nhiên

audio.addEventListener("ended", playRandomSong);

mobileAudio.addEventListener("ended", playRandomSong);

// Chữ đỏ chạy theo audio

let lyrics = [];

let currentLine = -1;

async function loadLyrics(lyricsUrl) {

try {

const response = await fetch(lyricsUrl);

if (!response.ok) throw new Error("Không thể tải lời bài hát");

const lrcText = await response.text();

parseLRC(lrcText);

} catch (error) {

console.error("Lỗi tải lời bài hát:", error);

document.getElementById("lyricsContainer").textContent =

"Lời bài hát chưa có.";

}

}

function parseLRC(lrcText) {

lyrics = lrcText

.split("\n")

.map((line) => {

const match = line.match(/\[(\d+):(\d+\.\d+)\](.\*)/);

if (match) {

const minutes = parseInt(match[1]);

const seconds = parseFloat(match[2]);

const time = minutes \* 60 + seconds;

return { time, text: match[3].trim() };

}

return null;

})

.filter((line) => line);

displayLyrics();

}

function displayLyrics() {

const lyricsContainer = document.getElementById("lyricsContainer");

lyricsContainer.innerHTML = lyrics

.map((lyric) => `<p>${lyric.text}</p>`)

.join("");

}

function highlightCurrentLyric(currentTime) {

for (let i = 0; i < lyrics.length; i++) {

if (lyrics[i].time > currentTime) {

if (currentLine !== i - 1) {

currentLine = i - 1;

updateLyricsHighlight();

}

break;

}

}

}

function updateLyricsHighlight() {

const lyricsContainer = document.getElementById("lyricsContainer");

const lines = lyricsContainer.getElementsByTagName("p");

Array.from(lines).forEach((line, index) => {

line.classList.toggle("active", index === currentLine);

});

}

// Sự kiện cập nhật lời theo thời gian của audio

audio.addEventListener("timeupdate", () => {

highlightCurrentLyric(audio.currentTime);

});

mobileAudio.addEventListener("timeupdate", () => {

highlightCurrentLyric(mobileAudio.currentTime);

});

// Cập nhật trong hàm displaySong để load lyrics

async function displaySong() {

if (songs.length === 0) return;

const song = songs[currentIndex];

document.getElementById("songTitle").textContent = song.title;

document.getElementById("songArtist").textContent = song.artist;

document.getElementById("audioPlayer").src = song.official;

document.getElementById("mobileAudioPlayer").src = song.official;

await loadLyrics(song.lyrics);

}

</script>

</body>

</html>