

Код примера Speech Audio Seach v2

[speech_audio_search_v2.php](#)

```
<?php
// PHP: Ищем все файлы с расширением .mp3 в папке music
$files = glob("music/*.mp3");
// Создаем пустой массив для хранения данных плейлиста
$playlist = [];
// Перебираем каждый найденный файл
foreach ($files as $file) {
    // Добавляем в массив: название трека (без расширения) и путь к файлу
    $playlist[] = ["title" => basename($file, ".mp3"), "src" => $file];
}
// Если папка пуста, создаем один пустой элемент, чтобы избежать ошибок
if (empty($playlist)) $playlist[] = ["title" => "Нет файлов", "src" =>
""];
// Переводим PHP-массив в формат JSON, чтобы его смог прочитать JavaScript
$jsonPlaylist = json_encode($playlist);
?>

<!DOCTYPE html>
<html lang="ru">
<head>
    <meta charset="UTF-8">
    <title>UCMS Voice Search Player</title>
    <style>
        /* Общие стили страницы: центрирование и светлый фон */
        body { background: #F0F2F5; display: flex; justify-content:
center; align-items: center; height: 100vh; margin: 0; font-family:
sans-serif; }
        /* Карточка плеера: фиксированный размер 600x300 в стиле Facebook */
        .fb-card { width: 600px; height: 300px; background: #fff;
border-radius: 8px; box-shadow: 0 12px 28px rgba(0,0,0,0.1); display:
flex; flex-direction: column; position: relative; border: 1px solid
#dddfe2; }
        /* Шапка плеера */
        .fb-header { padding: 10px 16px; border-bottom: 1px solid
#ebedf0; display: flex; justify-content: space-between; }
        /* Индикатор работы микрофона */
        .status-indicator { display: flex; align-items: center; gap:
5px; font-size: 12px; }
        .dot { width: 10px; height: 10px; border-radius: 50%;
background: #ccc; }
        /* Анимация пульсации для активного микрофона */
        .active-mic .dot { background: #31a24c; animation: pulse 1.5s
infinite; }
        @keyframes pulse { 0%, 100% { opacity: 1; } 50% { opacity: 0.4;
} }
```

```
/* Главная область с названием трека */
.fb-main { flex: 1; display: flex; flex-direction: column;
justify-content: center; align-items: center; padding: 20px; text-align: center; }
#currentTitle { font-size: 20px; font-weight: bold; color: #1c1e21; margin-bottom: 5px; cursor: pointer; }
/* Панель кнопок управления */
.fb-controls { display: flex; border-top: 1px solid #ebedf0; }
.fb-btn { flex: 1; background: none; border: none; padding: 12px; color: #65676B; font-weight: 600; cursor: pointer; }
/* Выдвижной плейлист (изначально высота 0) */
#playlistOverlay { position: absolute; bottom: 85px; left: 0; width: 100%; height: 0; background: #fff; overflow-y: auto; transition: 0.3s; z-index: 10; border-top: 1px solid #ddd; }
#playlistOverlay.open { height: 160px; } /* Раскрытое состояние плейлиста */
#playlistView { list-style: none; padding: 0; margin: 0; }
#playlistView li { padding: 10px 20px; border-bottom: 1px solid #eee; cursor: pointer; font-size: 13px; }
#playlistView li.active { color: #1877F2; background: #e7f3ff; font-weight: 600; }
/* Подсказки внизу плеера */
.hints { background: #f7f8fa; padding: 8px; font-size: 10px; color: #65676B; text-align: center; }
audio { display: none; } /* Плеер скрыт, так как мы используем свои кнопки */
</style>
</head>
<body>

<div class="fb-card" id="playerNode">
  <div class="fb-header">
    <b style="color:#1877F2">UCMS Audio Search</b>
    <!-- Статус микрофона -->
    <div class="status-indicator" id="statusBox"><div class="dot"></div> <span id="statusLabel">Кликните для активации</span></div>
  </div>
  <div class="fb-main">
    <!-- Текст текущего трека -->
    <div id="currentTitle" onclick="togglePlaylist()">Ожидание команды...</div>
    <!-- Слой для вывода распознанного текста -->
    <div id="debug" style="font-size: 11px; color: #1877F2; margin-top: 5px;"></div>
  </div>
  <!-- Контейнер плейлиста -->
  <div id="playlistOverlay"><ul id="playlistView"></ul></div>
  <!-- Кнопки управления -->
  <div class="fb-controls">
```

```
<button class="fb-btn" onclick="runCmd('назад')">Назад</button>
<button class="fb-btn" onclick="runCmd('играть')">Играть</button>
<button class="fb-btn" onclick="runCmd('пауза')">Стоп</button>
<button class="fb-btn"
onclick="runCmd('дальше')">Дальше</button>
</div>
<div class="hints">Скажите: <b>"Включи [название]"</b> или "играть",
"стоп", "дальше"</div>
<!-- Аудио-элемент HTML5 -->
<audio id="audio" crossorigin="anonymous"></audio>
</div>

<script>
// Переносим данные плейлиста из PHP в JS
const tracks = <?php echo $jsonPlaylist; ?>;
let currentIdx = 0; // Индекс текущего трека
const audio = document.getElementById('audio'); // Элемент аудио
const title = document.getElementById('currentTitle'); // Элемент
заголовка
const statusLabel = document.getElementById('statusLabel'); // Текст
статуса
const statusBox = document.getElementById('statusBox'); // Блок статуса
const debug = document.getElementById('debug'); // Блок отладки

// Функция синтеза речи (чтобы плеер отвечал голосом)
// Инициализируем объект синтеза речи
const synth = window.speechSynthesis;

function speak(txt) {
  if (!synth) return;
  synth.cancel(); // Прерываем старую речь

  // 1. Улучшаем текст для интонации:
  // Добавляем запятые и точки там, где нужны логические паузы
  // Например: "Включаю... [Название трека]. Приятного прослушивания."
  let processedText = txt.replace("Включаю", "Включаю,");
  if (!processedText.endsWith('.')) processedText += '.';

  const ut = new SpeechSynthesisUtterance(processedText);

  // 2. Настройка языка и поиск "человеческого" голоса
  ut.lang = 'ru-RU';

  // Получаем все голоса системы
  const voices = synth.getVoices();

  // Ищем наиболее качественные голоса (Microsoft или Google)
  // Эти голоса обычно звучат гораздо мягче и естественнее
  const premiumVoice = voices.find(v =>
    (v.name.includes('Google'))
    && v.lang.includes('ru'))
```

```
);

if (premiumVoice) {
    ut.voice = premiumVoice;
}

// 3. Тонкая настройка тембра (делаем голос менее "писклявым")
ut.rate = 1.0;    // Чуть замедляем (1.0 - быстро, 0.9 - размеренно и естественно)
ut.pitch = 1.1;   // Оставляем стандартную высоту (или 0.9 для более глубокого голоса)
ut.volume = 1.0;  // Громкость на максимум

// 4. Запуск озвучки
synth.speak(ut);
}

// Функция загрузки и запуска трека
function loadTrack(idx) {
    if (!tracks[idx] || !tracks[idx].src) return;

    currentIdx = idx;
    audio.src = tracks[idx].src;
    title.innerText = tracks[idx].title;

    document.querySelectorAll('#playlistView li').forEach((li, i) => {
        li.className = (i === idx) ? 'active' : '';
    });

    // Запускаем музыку
    audio.play().then(() => {
        // Запускаем голос ПОСЛЕ начала музыки (или одновременно)
        speak("Включаю " + tracks[idx].title);
    }).catch(err => {
        console.log("Браузер заблокировал звук до первого клика");
    });
}

// Главная функция обработки команд (кнопок и голоса)
function runCmd(cmd) {
    debug.innerText = "Вы сказали: " + cmd; // Вывод распознанного текста для проверки
    const text = cmd.toLowerCase(); // Приведение к нижнему регистру

    // Логика 1: Поиск песни по названию во всем плейлисте
    let foundIdx = -1;
    tracks.forEach((track, index) => {
        const trackTitle = track.title.toLowerCase();
        // Если в команде есть название трека или название содержит часть команды
```

```
        if (text.includes(trackTitle) || (text.includes('включи') &&
trackTitle.includes(text.replace('включи', '').trim())) {
            foundIdx = index;
        }
    });

    if (foundIdx !== -1) { // Если песня найдена
        loadTrack(foundIdx);
        return;
    }

    // Логика 2: Стандартные команды управления
    if (text.includes('играть') || text.includes('пуск')) {
audio.play(); speak("Запускаю");
    }
    else if (text.includes('стоп') || text.includes('пауза')) {
audio.pause(); speak("Пауза");
    }
    else if (text.includes('дальше')) loadTrack((currentIdx + 1) %
tracks.length);
    else if (text.includes('назад')) loadTrack((currentIdx - 1 +
tracks.length) % tracks.length);
    else if (text.includes('громче')) { audio.volume = Math.min(1,
audio.volume + 0.2); speak("Громче");
    }
    else if (text.includes('тише')) { audio.volume = Math.max(0,
audio.volume - 0.2); speak("Тише");
    }

    // Переключение видимости плейлиста
    function togglePlaylist() {
document.getElementById('playlistOverlay').classList.toggle('open');
    }

    // Инициализация распознавания речи
    const SpeechRecognition = window.SpeechRecognition ||
window.webkitSpeechRecognition;
    if (SpeechRecognition) {
        const recognition = new SpeechRecognition();
        recognition.lang = 'ru-RU';
        recognition.continuous = false; // Отключаем постоянный режим для
лучшей точности

        // Когда браузер распознал фразу
        recognition.onresult = (e) => {
            const transcript = e.results[0][0].transcript; // Получаем текст
            runCmd(transcript); // Отправляем в обработчик
        };

        // Когда микрофон отключается (после фразы или тишины)
        recognition.onend = () => {
            // Если плеер в этот момент не "разговаривает", включаем микрофон снова
            if (!window.speechSynthesis.speaking) {
                recognition.start();
            } else {

```

```
// Если плеер говорит, ждем секунду и пробуем снова
    setTimeout(() => recognition.start(), 1000);
  }
};

// Обработка ошибок микрофона
recognition.onerror = (e) => {
  if (e.error === 'no-speech') console.log("Тишина...");
};

// Активация микрофона по клику (требование безопасности браузеров)
document.body.onclick = () => {
  try {
    recognition.start();
    statusBox.classList.add('active-mic'); // Визуальный эффект
    statusLabel.innerText = "Слушаю название...";
    if(!audio.src) loadTrack(0); // Загружаем первый трек при
первом клике
  } catch(err) { /* Игнорируем ошибку, если уже запущен */ }
};

// Формируем список песен в HTML из данных плейлиста
const list = document.getElementById('playlistView');
list.innerHTML = tracks.map((t, i) => `- 

```

From:
<https://wwoss.direct.quickconnect.to/> - worldwide open-source software

Permanent link:
https://wwoss.direct.quickconnect.to/doku.php?id=software:development:demo:cms:ucms:appendix:appendix_js_speech_audio_search_v2_comment

Last update: 2026/05/07 10:27

