<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Login - Book My Slot</title>

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

    <link rel="icon" href="data:;base64,iVBORw0KGgo="> <!-- Prevent favicon error -->

</head>

<body>

{{> header }}

{{> headers }}

<div class="main-container">

    <!-- Buttons to show login forms -->

    <div class="button-container">

        <button id="groundOwnerLoginBtn">LOGIN AS GROUND OWNER</button>

        <button id="teamLoginBtn">LOGIN AS TEAM</button>

    </div>

    <!-- Login as Ground Owner -->

    <div class="login-box" id="groundOwnerLoginForm" style="display:none;">

        <h2>LOGIN AS GROUND OWNER</h2>

        <form id="groundOwnerForm">

            <input type="email" id="groundOwnerEmail" placeholder="Enter Your Email" required>

            <input type="password" id="groundOwnerPassword" placeholder="Enter Your Password" required>

            <button type="submit">LOGIN</button>

        </form>

        <p>Don't have an account? <a href="/signup">Sign up</a></p>

    </div>

    <!-- Login as Team -->

    <div class="login-box" id="teamLoginForm" style="display:none;">

        <h2>LOGIN AS TEAM</h2>

        <form id="teamForm">

            <input type="email" id="teamEmail" placeholder="Enter Captain's Email" required>

            <input type="password" id="teamPassword" placeholder="Enter Your Password" required>

            <button type="submit">LOGIN</button>

        </form>

        <p>Don't have an account? <a href="/signup">Sign up</a></p>

    </div>

</div>

<script>

document.addEventListener("DOMContentLoaded", function () {

    console.log("🚀 Script Loaded Successfully!");

    // ✅ Select Forms & Buttons

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

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

    const groundOwnerBtn = document.getElementById("groundOwnerLoginBtn");

    const teamOwnerBtn = document.getElementById("teamLoginBtn");

    console.log("✅ Checking if elements exist...");

    console.log("🔹 Ground Owner Form:", groundOwnerForm);

    console.log("🔹 Team Form:", teamForm);

    console.log("🔹 Ground Owner Button:", groundOwnerBtn);

    console.log("🔹 Team Owner Button:", teamOwnerBtn);

    if (!groundOwnerForm) console.error("❌ groundOwnerForm NOT found!");

    if (!teamForm) console.error("❌ teamForm NOT found!");

    if (!groundOwnerBtn) console.error("❌ groundOwnerBtn NOT found!");

    if (!teamOwnerBtn) console.error("❌ teamOwnerBtn NOT found!");

    // ✅ Add Click Listeners for "Login as" Buttons

    if (groundOwnerBtn) {

        groundOwnerBtn.addEventListener("click", function () {

            console.log("🟢 BUTTON CLICKED: Ground Owner Login");

            showForm("groundOwnerLoginForm");

        });

    }

    if (teamOwnerBtn) {

        teamOwnerBtn.addEventListener("click", function () {

            console.log("🟢 BUTTON CLICKED: Team Owner Login");

            showForm("teamLoginForm");

        });

    }

    function showForm(formId) {

        console.log(`🟢 Showing form: ${formId}`);

        document.getElementById("groundOwnerLoginForm").style.display = "none";

        document.getElementById("teamLoginForm").style.display = "none";

        document.getElementById(formId).style.display = "block";

    }

    // ✅ Attach event listeners to login forms

    if (groundOwnerForm) {

        groundOwnerForm.addEventListener("submit", function (e) {

            e.preventDefault();

            console.log("🟢 CLICK: Ground Owner Login Button Pressed!");

            loginUser("Ground Owner", "groundOwnerEmail", "groundOwnerPassword");

        });

    }

    if (teamForm) {

        teamForm.addEventListener("submit", function (e) {

            e.preventDefault();

            console.log("🟢 CLICK: Team Owner Login Button Pressed!");

            loginUser("Team Owner", "teamEmail", "teamPassword");

        });

    }

    // ✅ Login User Function

    async function loginUser(role, emailId, passwordId) {

        const email = document.getElementById(emailId).value;

        const password = document.getElementById(passwordId).value;

        console.log(`🔍 Logging in as: ${role}`);

        console.log(`📧 Email: ${email}`);

        console.log(`🔑 Password: ${password}`);

        try {

            console.log("🟢 Sending login request to /api/auth/login...");

            const response = await fetch("/api/auth/login", {

                method: "POST",

                headers: { "Content-Type": "application/json" },

                body: JSON.stringify({ email, password }),

            });

            const data = await response.json();

            console.log("🔍 Login Response:", data);

            if (response.ok) {

                // Save token and user data to localStorage

                localStorage.setItem("token", data.token);

                localStorage.setItem("user", JSON.stringify(data.user));

                console.log(`✅ ${role} Logged In Successfully`);

                console.log("🔑 Token saved to localStorage:", data.token);

                console.log("👤 User data saved to localStorage:", data.user);

                // Redirect if backend provides a URL

                if (data.redirect) {

                    console.log(`✅ Redirecting to: ${data.redirect}`);

                    window.location.href = data.redirect;

                } else {

                    console.log("⚠️ No redirect URL provided.");

                    alert("Login successful, but no redirection provided.");

                    updateUI(true);

                }

            } else {

                console.error("❌ Login failed:", data.message);

                alert(data.message);

            }

        } catch (error) {

            console.error("❌ Login Error:", error);

            alert("An error occurred. Please try again.");

        }

    }

    // ✅ Update UI Function

    function updateUI(isLoggedIn) {

        console.log(`🟢 Updating UI. Logged in: ${isLoggedIn}`);

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

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

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

        if (isLoggedIn) {

            console.log("🟢 Hiding login/signup buttons, showing logout button.");

            loginBtn.style.display = "none";

            signupBtn.style.display = "none";

            logoutBtn.style.display = "block";

        } else {

            console.log("🟢 Showing login/signup buttons, hiding logout button.");

            loginBtn.style.display = "inline-block";

            signupBtn.style.display = "inline-block";

            logoutBtn.style.display = "none";

        }

    }

    // ✅ Check Authentication Status

    async function checkAuthStatus() {

        try {

            console.log("🔍 Checking authentication status...");

            const token = localStorage.getItem("token");

            if (!token) {

                console.log("❌ No token found in localStorage");

                return updateUI(false);

            }

            console.log("🔑 Token found in localStorage:", token);

            const response = await fetch('/api/auth/status', {

                method: 'GET',

                headers: {

                    'Content-Type': 'application/json',

                    'Authorization': `Bearer ${token}` // Include token in the header

                },

                credentials: 'include'

            });

            const data = await response.json();

            console.log("🔍 Auth Check Response:", data);

            if (data.isAuthenticated) {

                console.log("✅ User is authenticated.");

            } else {

                console.log("❌ User is NOT authenticated.");

            }

            updateUI(data.isAuthenticated);

        } catch (error) {

            console.error("⚠️ Error checking authentication status:", error);

        }

    }

    checkAuthStatus();

});

</script>

</body>

</html>