cartella progetto: F:landing-page-libri

progetto Vercel: catalogo-libri-nuovo

immagine sito attuale: Immagine che contiene testo, schermata, design

Il contenuto generato dall'IA potrebbe non essere corretto.

database Firebae: aggiungilibri

admin.css

/\* Stile generale \*/

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

margin: 0;

padding: 20px;

text-align: center;

}

/\* Titoli \*/

h1 {

color: #333;

font-size: 24px;

}

h2 {

color: #555;

margin-top: 30px;

font-size: 20px;

}

/\* Contenitore del form \*/

form {

width: 100%;

max-width: 400px;

margin: 20px auto;

padding: 20px;

background: white;

border-radius: 8px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

/\* Etichette e input \*/

label {

display: block;

margin-top: 10px;

font-weight: bold;

font-size: 14px;

text-align: left;

}

input[type="text"],

input[type="url"],

textarea,

input[type="number"],

input[type="file"],

select {

width: 100%;

padding: 10px;

margin-top: 5px;

border: 1px solid #ddd;

border-radius: 5px;

font-size: 14px;

}

/\* Correzione allineamento caselle di spunta \*/

.formati-container {

display: flex;

gap: 10px;

flex-wrap: wrap;

justify-content: center;

margin-top: 10px;

}

.formati-container label {

display: flex;

align-items: center;

gap: 5px;

font-size: 14px;

}

/\* Stile pulsanti \*/

button {

width: 100%;

background-color: #28a745;

color: white;

padding: 12px;

margin-top: 15px;

border: none;

border-radius: 5px;

cursor: pointer;

font-size: 16px;

transition: 0.3s;

}

button:hover {

background-color: #218838;

}

/\* Contenitore libri inseriti \*/

#libri-inseriti {

margin-top: 30px;

display: flex;

flex-wrap: wrap;

justify-content: center;

gap: 15px;

}

/\* Box singolo per ogni libro \*/

.admin-libro {

width: 180px;

padding: 15px;

background: white;

border-radius: 8px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

text-align: center;

transition: transform 0.3s;

}

.admin-libro:hover {

transform: scale(1.05);

}

.admin-libro img {

width: 100px;

height: 150px;

object-fit: cover;

border-radius: 5px;

}

/\* Pulsante elimina libro \*/

.admin-libro button {

width: auto;

padding: 8px 12px;

background-color: #dc3545;

border: none;

color: white;

font-size: 12px;

cursor: pointer;

border-radius: 4px;

margin-top: 10px;

}

.admin-libro button:hover {

background-color: #c82333;

}

admin.html

<!DOCTYPE html>

<html lang="it">

<head>

<meta charset="UTF-8">

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

<title>Pannello di amministrazione</title>

<link rel="stylesheet" href="admin.css">

</head>

<body>

<h1>Pannello di amministrazione</h1>

<div id="login-container">

<button id="login-button">Accedi con Google</button>

</div>

<div id="admin-panel" style="display: none;">

<button id="logout-button">Esci</button>

<h2>Inserisci nuovo libro</h2>

<form id="libro-form">

<label for="titolo">Titolo:</label>

<input type="text" id="titolo" name="titolo" required>

<label for="autore">Autore:</label>

<input type="text" id="autore" name="autore" required>

<label for="descrizione">Descrizione:</label>

<textarea id="descrizione" name="descrizione" required></textarea>

<label for="linkAmazon">Link Amazon:</label>

<input type="url" id="linkAmazon" name="linkAmazon" required>

<label for="immagine">Immagine di copertina:</label>

<input type="file" id="immagine" name="immagine" accept="image/\*">

<label for="prezzo">Prezzo:</label>

<select id="valuta" name="valuta">

<option value="$">$</option>

<option value="€">€</option>

<option value="£">£</option>

</select>

<input type="number" id="prezzo" name="prezzo" step="0.01" required>

<label for="formati">Formati disponibili:</label><br>

<input type="checkbox" id="ebook" name="formati" value="eBook">

<label for="ebook">eBook</label>

<input type="checkbox" id="paperback" name="formati" value="Paperback">

<label for="paperback">Paperback</label>

<input type="checkbox" id="hardcover" name="formati" value="Hardcover">

<label for="hardcover">Hardcover</label>

<button type="submit">Aggiungi libro</button>

</form>

<h2>Filtra e ordina i libri</h2>

<div id="filtro-container">

<input type="text" id="filtro-testo" placeholder="Cerca per titolo o autore...">

<select id="ordinamento">

<option value="titolo">Ordina per Titolo</option>

<option value="autore">Ordina per Autore</option>

<option value="prezzo">Ordina per Prezzo</option>

</select>

</div>

<h2>Backup e Ripristino</h2>

<button id="esporta-dati">Esporta JSON</button>

<input type="file" id="importa-dati" accept=".json">

<button id="carica-dati">Importa JSON</button>

<h2>Libri inseriti</h2>

<div id="libri-inseriti"></div>

</div>

<script src="https://www.gstatic.com/firebasejs/9.6.1/firebase-app-compat.js"></script>

<script src="https://www.gstatic.com/firebasejs/9.6.1/firebase-auth-compat.js"></script>

<script src="https://www.gstatic.com/firebasejs/9.6.1/firebase-database-compat.js"></script>

<script src="https://www.gstatic.com/firebasejs/9.6.1/firebase-storage-compat.js"></script>

<script src="admin.js"></script>

</body>

</html>

admin.js

// LISTA AMMINISTRATORI AUTORIZZATI

const amministratoriAutorizzati = [

"robpacpublishing@gmail.com"

];

// ELEMENTI HTML

const loginButton = document.getElementById("login-button");

const logoutButton = document.getElementById("logout-button");

const loginContainer = document.getElementById("login-container");

const adminPanel = document.getElementById("admin-panel");

const libroForm = document.getElementById("libro-form");

const libriInseriti = document.getElementById("libri-inseriti");

const filtroTesto = document.getElementById("filtroTesto") || null;

const ordinamento = document.getElementById("ordinamento") || null;

// CACHE LOCALE PER MEMORIZZARE I LIBRI

let cacheLibri = [];

// CREAZIONE BOX NOTIFICHE

const notifica = document.createElement("div");

notifica.id = "notifica";

document.body.appendChild(notifica);

// MOSTRA NOTIFICA

function mostraNotifica(testo, tipo = "successo") {

notifica.textContent = testo;

notifica.className = tipo === "errore" ? "notifica-errore" : "notifica-successo";

notifica.style.display = "block";

setTimeout(() => {

notifica.style.display = "none";

}, 5000);

}

// CONTROLLO ACCESSO AMMINISTRATORI

firebase.auth().onAuthStateChanged(user => {

if (user) {

if (amministratoriAutorizzati.includes(user.email)) {

loginContainer.style.display = "none";

adminPanel.style.display = "block";

caricaLibri();

} else {

mostraNotifica("⚠️ Accesso negato. Non sei un amministratore!", "errore");

firebase.auth().signOut();

}

} else {

loginContainer.style.display = "block";

adminPanel.style.display = "none";

}

});

// LOGIN CON GOOGLE

loginButton.addEventListener("click", () => {

const provider = new firebase.auth.GoogleAuthProvider();

firebase.auth().signInWithPopup(provider)

.then((result) => {

const user = result.user;

if (user) {

mostraNotifica(`Benvenuto ${user.displayName}!`, "successo");

setTimeout(() => {

location.reload();

}, 1000);

}

})

.catch(error => {

console.error("Errore di autenticazione:", error);

mostraNotifica("Errore di autenticazione: " + error.message, "errore");

});

});

// LOGOUT

logoutButton.addEventListener("click", () => {

firebase.auth().signOut().then(() => {

mostraNotifica("Logout effettuato con successo", "successo");

location.reload();

}).catch(error => mostraNotifica("Errore nel logout", "errore"));

});

// CARICAMENTO LIBRI

function caricaLibri() {

firebase.database().ref('libri').once('value').then(snapshot => {

cacheLibri = snapshot.val() ? Object.entries(snapshot.val()).map(([id, libro]) => ({ id, ...libro })) : [];

mostraLibriInseriti();

}).catch(error => mostraNotifica("Errore nel caricamento dei libri: " + error.message, "errore"));

}

// FUNZIONE UPLOAD CLOUDINARY

async function uploadImmagine(file) {

const formData = new FormData();

formData.append("file", file);

formData.append("upload\_preset", "robpac\_upload"); // Il tuo preset

formData.append("folder", "copertine"); // La cartella specificata in Cloudinary

try {

const response = await fetch("https://api.cloudinary.com/v1\_1/robpac/image/upload", {

method: "POST",

body: formData

});

if (!response.ok) {

throw new Error(`Errore nell'upload: ${response.statusText}`);

}

const data = await response.json();

return data.secure\_url;

} catch (error) {

console.error("Errore nell'upload su Cloudinary:", error);

return null;

}

}

// INSERIMENTO NUOVO LIBRO

document.getElementById("libro-form").addEventListener("submit", async function(event) {

event.preventDefault(); // Evita il refresh della pagina

const titolo = document.getElementById("titolo").value.trim();

const autore = document.getElementById("autore").value.trim();

const descrizione = document.getElementById("descrizione").value.trim();

const linkAmazon = document.getElementById("linkAmazon").value.trim();

const immagine = document.getElementById("immagine").files[0];

const prezzoInput = document.getElementById("prezzo").value.trim();

const prezzo = parseFloat(prezzoInput.replace(",", "."));

if (isNaN(prezzo)) {

return mostraNotifica("⚠️ Inserisci un prezzo valido!", "errore");

}

const valuta = document.getElementById("valuta").value;

if (!titolo || !autore || !descrizione || !linkAmazon) {

return mostraNotifica("⚠️ Tutti i campi devono essere compilati!", "errore");

}

// \*\*VERIFICA DUPLICATI\*\*

const libroEsistente = cacheLibri.find(libro => libro.titolo.toLowerCase() === titolo.toLowerCase() && libro.autore.toLowerCase() === autore.toLowerCase());

if (libroEsistente) {

return mostraNotifica("⚠️ Questo libro è già stato inserito!", "errore");

}

let urlImmagine = "placeholder.jpg"; // Se l'immagine non è caricata, usa un placeholder

// SE È STATA CARICATA UN'IMMAGINE, LA MEMORIZZA IN CLOUDINARY

if (immagine) {

urlImmagine = await uploadImmagine(immagine);

if (!urlImmagine) {

return mostraNotifica("Errore nel caricamento immagine!", "errore");

}

}

// CREA L'OGGETTO LIBRO

const libro = {

titolo,

autore,

descrizione,

linkAmazon,

prezzo: prezzo.toFixed(2), // Ora accetta due decimali

valuta,

immagine: urlImmagine

};

// AGGIUNGE AL DATABASE

const libriRef = firebase.database().ref('libri').push();

libro.id = libriRef.key;

libriRef.set(libro)

.then(() => {

cacheLibri.push(libro);

mostraLibriInseriti();

mostraNotifica("📚 Libro aggiunto con successo!", "successo");

document.getElementById("libro-form").reset();

})

.catch(error => {

mostraNotifica("Errore durante l'aggiunta del libro: " + error.message, "errore");

});

});

// MOSTRA LIBRI INSERITI CON MINIATURA

function mostraLibriInseriti() {

libriInseriti.innerHTML = "";

cacheLibri.forEach(libro => {

const libroDiv = document.createElement("div");

libroDiv.classList.add("admin-libro");

libroDiv.innerHTML = `

<h3>${libro.titolo}</h3>

<p>${libro.autore}</p>

${libro.immagine ? `<img src="${libro.immagine}" alt="${libro.titolo}" style="width: 100px; height: 150px; object-fit: cover; border-radius: 5px;">` : ''}

<button onclick="eliminaLibro('${libro.id}')">🗑️ Elimina</button>

`;

libriInseriti.appendChild(libroDiv);

});

}

// ELIMINA LIBRO

function eliminaLibro(libroId) {

firebase.database().ref('libri/' + libroId).remove()

.then(() => {

cacheLibri = cacheLibri.filter(libro => libro.id !== libroId);

mostraLibriInseriti();

mostraNotifica("Libro eliminato con successo!", "successo");

})

.catch(error => mostraNotifica("Errore nell'eliminazione del libro: " + error.message, "errore"));

}

// EVENTI FILTRO E ORDINAMENTO

if (filtroTesto) filtroTesto.addEventListener("input", mostraLibriInseriti);

if (ordinamento) ordinamento.addEventListener("change", mostraLibriInseriti);

index.html

<!DOCTYPE html>

<html lang="it">

<head>

<meta charset="UTF-8">

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

<title>Catalogo Libri</title>

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

</head>

<body>

<header>

<div id="titolo-container">

<h1>RobPac Publishing</h1>

<h2>A World of Books at Your Fingertips</h2>

</div>

<div id="logo-container">

<img id="logo-catalogo" src="logo3senzasfondo.png" alt="Logo Catalogo">

</div>

<img id="banner-catalogo" src="banner-landpage2.png" alt="Banner Catalogo">

</header>

<div id="login-container">

<button id="login-button">Accedi con Google</button>

</div>

<div id="admin-panel" style="display: none;">

<button id="logout-button">Logout</button>

<h2>Gestione Libri</h2>

<form id="libro-form">

<input type="text" id="titolo" placeholder="Titolo" required>

<input type="text" id="autore" placeholder="Autore" required>

<textarea id="descrizione" placeholder="Descrizione" required></textarea>

<input type="url" id="linkAmazon" placeholder="Link Amazon" required>

<input type="file" id="immagine" accept="image/\*">

<input type="number" id="prezzo" placeholder="Prezzo" step="0.01" min="0.01" required>

<select id="valuta">

<option value="$">Dollaro ($)</option>

<option value="€">Euro (€)</option>

</select>

<button type="submit">Aggiungi Libro</button>

</form>

<h2>Libri Inseriti</h2>

<!-- 🔹 Aggiungi il campo di ricerca prima della lista dei libri -->

<input type="text" id="filtro-testo" placeholder="Cerca libri...">

<div id="libri-inseriti"></div>

</div>

<!-- Aggiunta di elementi mancanti per evitare errori JS -->

<div id="libri"></div>

<div id="info-container"></div>

<footer>

<p>&copy; 2025 Rob Pac - All Rights Reserved</p>

</footer>

<script src="https://www.gstatic.com/firebasejs/9.6.1/firebase-app-compat.js"></script>

<script src="https://www.gstatic.com/firebasejs/9.6.1/firebase-auth-compat.js"></script>

<script src="https://www.gstatic.com/firebasejs/9.6.1/firebase-database-compat.js"></script>

<script src="https://www.gstatic.com/firebasejs/9.6.1/firebase-storage-compat.js"></script>

<script src="script.js"></script>

<script src="admin.js"></script>

</body>

</html>

libri.json

[]

script.js

const firebaseConfig = {

apiKey: "AIzaSyAOIp2reVVoeikYjZUk73yQpZNPaDVvCkw",

authDomain: "aggiungilibri.firebaseapp.com",

databaseURL: "https://aggiungilibri-default-rtdb.firebaseio.com",

projectId: "aggiungilibri",

storageBucket: "aggiungilibri.firebasestorage.app",

messagingSenderId: "215130413037",

appId: "1:215130413037:web:058d3395ddef3b7441f9e4"

};

firebase.initializeApp(firebaseConfig);

const libriDiv = document.getElementById("libri");

const infoContainer = document.getElementById("info-container");

if (!libriDiv || !infoContainer) {

console.error("Errore: Elementi HTML mancanti. Verifica che esistano in index.html");

}

function mostraLibri() {

const libriRef = firebase.database().ref('libri');

libriRef.once('value').then(snapshot => {

const libri = snapshot.val();

libriDiv.innerHTML = "";

if (libri) {

Object.entries(libri).forEach(([id, libro]) => {

const libroDiv = document.createElement("div");

libroDiv.classList.add("section-libro");

const immagine = document.createElement("img");

immagine.src = libro.immagine || "placeholder.jpg"; // Se manca l'immagine, mostra un placeholder

immagine.alt = libro.titolo;

const titolo = document.createElement("h3");

titolo.textContent = libro.titolo;

const autore = document.createElement("p");

autore.textContent = "di " + libro.autore;

const prezzo = document.createElement("p");

prezzo.innerHTML = "<b>" + libro.valuta + libro.prezzo + "</b>";

const buyButton = document.createElement("button");

buyButton.classList.add("buy-now-button");

buyButton.textContent = "Compra ora";

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

window.open(libro.linkAmazon, "\_blank");

});

libroDiv.appendChild(immagine);

libroDiv.appendChild(titolo);

libroDiv.appendChild(autore);

libroDiv.appendChild(prezzo);

libroDiv.appendChild(buyButton);

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

mostraInfoLibro(id);

});

libriDiv.appendChild(libroDiv);

});

}

}).catch(error => {

console.error("Errore nel recupero dei libri:", error);

});

}

function mostraInfoLibro(libroId) {

const libriRef = firebase.database().ref(`libri/${libroId}`);

libriRef.once('value').then(snapshot => {

const libro = snapshot.val();

if (!libro) return;

const infoDiv = document.createElement("div");

infoDiv.classList.add("info-libro");

const closeButton = document.createElement("span");

closeButton.classList.add("close-button");

closeButton.innerHTML = "&times;";

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

infoContainer.innerHTML = "";

});

const immagine = document.createElement("img");

immagine.src = libro.immagine || "placeholder.jpg";

immagine.alt = libro.titolo;

const titolo = document.createElement("h3");

titolo.textContent = libro.titolo;

const autore = document.createElement("p");

autore.textContent = "di " + libro.autore;

const descrizione = document.createElement("p");

descrizione.textContent = libro.descrizione;

const prezzo = document.createElement("p");

prezzo.innerHTML = "<b>" + libro.valuta + libro.prezzo + "</b>";

const formati = document.createElement("p");

formati.textContent = "Formati disponibili: " + (libro.formati ? libro.formati.join(", ") : "N/A");

const buyButton = document.createElement("button");

buyButton.classList.add("buy-now-button");

buyButton.textContent = "Compra ora";

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

window.open(libro.linkAmazon, "\_blank");

});

infoDiv.appendChild(closeButton);

infoDiv.appendChild(immagine);

infoDiv.appendChild(titolo);

infoDiv.appendChild(autore);

infoDiv.appendChild(descrizione);

infoDiv.appendChild(prezzo);

infoDiv.appendChild(formati);

infoDiv.appendChild(buyButton);

infoContainer.innerHTML = "";

infoContainer.appendChild(infoDiv);

}).catch(error => {

console.error("Errore nel recupero delle informazioni del libro:", error);

});

}

function mostraLogo() {

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

if (logo) {

document.getElementById("logo-catalogo").src = logo;

}

}

function mostraBanner() {

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

if (banner) {

document.getElementById("banner-catalogo").src = banner;

}

}

mostraLibri();

mostraLogo();

mostraBanner();

style.css

body {

font-family: sans-serif;

margin: 0;

padding: 0;

background-color: #f0f8ff;

}

/\* HEADER \*/

header {

background-color: #f0f0f0;

padding: 20px;

text-align: center;

position: relative;

}

#titolo-container {

position: absolute;

top: 10px;

left: 50%;

transform: translateX(-50%);

z-index: 1;

}

#logo-container {

position: absolute;

top: 50%;

left: 20px;

transform: translateY(-50%);

z-index: 1;

}

#logo-catalogo {

max-height: 250px;

}

#banner-catalogo {

width: 100%;

max-height: 300px;

object-fit: cover;

}

/\* CONTENUTO PRINCIPALE \*/

main {

padding: 20px;

}

/\* FORM GESTIONE LIBRI \*/

#admin-panel {

max-width: 900px;

margin: 20px auto;

padding: 20px;

background: #f5f5f5;

border-radius: 8px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

/\* FORM IN GRIGLIA \*/

#libro-form {

display: grid;

grid-template-columns: repeat(2, 1fr);

gap: 15px;

align-items: center;

}

#libro-form label {

font-weight: bold;

}

#libro-form input,

#libro-form textarea,

#libro-form select,

#libro-form button {

width: 100%;

padding: 8px;

border: 1px solid #ccc;

border-radius: 4px;

font-size: 14px;

}

/\* LABEL CHIARA PER UPLOAD IMMAGINE \*/

#libro-form #label-immagine {

grid-column: span 2;

font-weight: bold;

text-align: center;

margin-top: 10px;

}

/\* TESTO DI SELEZIONE IMMAGINE \*/

#libro-form #immagine {

width: 100%;

display: block;

padding: 5px;

font-size: 14px;

cursor: pointer;

}

/\* MENU A TENDINA VALUTE \*/

#valuta {

width: 100%;

}

/\* MENU VALUTE CON OPZIONI RIORDINATE \*/

#valuta option:first-child {

font-weight: bold;

}

/\* BOTTONE DI INVIO \*/

#libro-form button {

grid-column: span 2;

background: #007bff;

color: white;

border: none;

cursor: pointer;

font-weight: bold;

}

#libro-form button:hover {

background: #0056b3;

}

/\* LIBRI INSERITI \*/

#libri-inseriti {

margin-top: 20px;

display: flex;

flex-wrap: wrap;

gap: 15px;

}

.admin-libro {

background: white;

padding: 10px;

border: 1px solid #ddd;

border-radius: 5px;

text-align: center;

box-shadow: 0px 2px 5px rgba(0, 0, 0, 0.1);

width: 200px;

}

.admin-libro img {

width: 100%;

height: auto;

border-radius: 5px;

object-fit: cover;

}

/\* PULSANTE ELIMINAZIONE LIBRI \*/

.admin-libro button {

background: #ff4d4d;

color: white;

padding: 5px 10px;

border: none;

border-radius: 4px;

cursor: pointer;

font-size: 12px;

margin-top: 5px;

}

.admin-libro button:hover {

background: #cc0000;

}

/\* FOOTER \*/

footer {

text-align: center;

padding: 10px;

background: #f0f0f0;

font-size: 14px;

}

/\* ADATTAMENTO MOBILE \*/

@media (max-width: 768px) {

#logo-catalogo {

max-height: 100px;

}

#banner-catalogo {

max-height: 200px;

}

#libri {

grid-template-columns: repeat(auto-fit, minmax(120px, 1fr));

gap: 10px;

}

.section-libro img {

width: 120px;

height: 180px;

}

.info-libro {

flex-direction: column;

width: 100%;

}

.info-libro img {

width: 100%;

margin-right: 0;

margin-bottom: 20px;

}

#titolo-container h1 {

font-size: 20px;

}

#titolo-container h2 {

font-size: 16px;

}

#admin-panel {

width: 95%;

padding: 10px;

}

#libro-form {

grid-template-columns: 1fr;

}

#libro-form textarea,

#libro-form button {

grid-column: span 1;

}

}

cors.json

[

{

"origin": ["http://localhost:8000"],

"method": ["GET", "POST", "PUT", "DELETE", "OPTIONS"],

"maxAgeSeconds": 3600

}

]