|  |
| --- |
| **tinc un fitxer de personal:**  **CREATE TABLE personal(**  **id\_personal INTEGER PRIMARY KEY AUTO\_INCREMENT**  **,nom VARCHAR(50) NOT NULL**  **,rol VARCHAR(15) NOT NULL CHECK(rol IN ('cuiner', 'cambrer', 'administrador'))**  **,username VARCHAR(15) NOT NULL**  **,password VARCHAR(15) NOT NULL**  **,pwdhash CHAR(64) NOT NULL**  **,host VARCHAR(15) NOT NULL**  **,es\_actiu BOOLEAN NOT NULL DEFAULT TRUE**  **,UNIQUE (username, host)**  **);**  **i els inserts els faig amb ed25519\_password('cuiner2'), per exemple:**  **INSERT INTO personal (nom, rol, username, password, pwdhash, host) VALUES('Kevin José CUI', 'cuiner', 'cuiner2', 'cuiner2', ed25519\_password('cuiner2'), 'localhost');**  **INSERT INTO personal (nom, rol, username, password, pwdhash, host) VALUES('Natàlia CUI', 'cuiner', 'cuiner3', 'cuiner3', ed25519\_password('cuiner3'), 'localhost');**  **com faig un login amb els camps:**  **username**  **password**  **host**  **tenint en compte el ed25519\_password()?**  **ChatGPT ha dit:** |

Per implementar un **login** tenint en compte que estàs utilitzant la funció ed25519\_password() per generar un hash de la contrasenya (com el camp pwdhash), has de fer el següent:

**🧠 Punts clau a tenir en compte**

1. **No has de desar la contrasenya en clar (password) a la base de dades**, perquè és un risc de seguretat molt alt. El camp password hauria de ser eliminat si ja tens pwdhash.
2. La funció ed25519\_password() **no és una funció estàndard** de MySQL o PHP. Si és pròpia o afegida via un plugin, necessitaràs assegurar-te que també pots usar-la des de PHP, o bé reemplaçar-la amb una funció com password\_hash() i password\_verify() en PHP.

**✅ Solució recomanada en PHP**

Suposant que tens el hash a pwdhash, i **pots fer servir la mateixa funció ed25519\_password()** des de SQL (o en el pitjor dels casos, des de PHP si la implementes), pots fer la consulta així:

<?php

// Connexió a la base de dades

$conn = new mysqli("localhost", "usuari", "contrasenya", "basedades");

// Recollir dades del formulari

$username = $\_POST['username'];

$password = $\_POST['password'];

$host = $\_POST['host'];

// Escapar valors per seguretat (si no fas servir consultes preparades)

$username = $conn->real\_escape\_string($username);

$password = $conn->real\_escape\_string($password);

$host = $conn->real\_escape\_string($host);

// Consulta amb ed25519\_password

$sql = "SELECT \* FROM personal

WHERE username = '$username'

AND host = '$host'

AND pwdhash = ed25519\_password('$password')

AND es\_actiu = 1";

$result = $conn->query($sql);

if ($result && $result->num\_rows === 1) {

// Login correcte

$usuari = $result->fetch\_assoc();

echo "Benvingut, " . $usuari['nom'];

// Guardar dades a la sessió si cal

} else {

// Login incorrecte

echo "Credencials incorrectes.";

}

$conn->close();

?>

**🔐 Recomanació de seguretat**

Tot i que utilitzes ed25519\_password(), et recomano valorar aquesta alternativa més estàndard amb PHP:

**En el moment d'inserir:**

$pwdhash = password\_hash($password, PASSWORD\_DEFAULT);

**En el login:**

$sql = "SELECT \* FROM personal WHERE username = ? AND host = ? AND es\_actiu = 1";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("ss", $username, $host);

$stmt->execute();

$result = $stmt->get\_result();

if ($row = $result->fetch\_assoc()) {

if (password\_verify($password, $row['pwdhash'])) {

echo "Benvingut " . $row['nom'];

} else {

echo "Contrasenya incorrecta";

}

} else {

echo "Usuari no trobat";

}

Vols que t’ajudi a migrar cap a password\_hash() o et cal continuar amb ed25519\_password() perquè tens requeriments específics?

|  |
| --- |
| **puc migrar cap a password\_hash() tenint en compte que la base de dades és mariadb?** |

Sí, **pots migrar perfectament cap a password\_hash() encara que la teva base de dades sigui MariaDB**, perquè:

👉 **password\_hash() i password\_verify() són funcions de PHP**, no de la base de dades.  
➡️ Això vol dir que el **hash es genera i comprova en PHP**, i **MariaDB només guarda la cadena resultant** com un simple CHAR(255) o VARCHAR(255).

**✅ Com migrar correctament**

**1. 🔧 Canvia el camp pwdhash a un tipus més gran (opcional però recomanat)**

Els hashes generats per password\_hash() poden arribar als **255 caràcters**. Modifica la teva taula així:

ALTER TABLE personal MODIFY pwdhash VARCHAR(255) NOT NULL;

**2. 🔒 Elimina el camp password (opcional però recomanat)**

Per seguretat, si ja tens el hash, aquest camp és innecessari:

ALTER TABLE personal DROP COLUMN password;

**3. 🧑‍🍳 Inserció d'usuaris nous des de PHP**

$nom = "Kevin José CUI";

$rol = "cuiner";

$username = "cuiner2";

$password = "cuiner2";

$host = "localhost";

$pwdhash = password\_hash($password, PASSWORD\_DEFAULT); // genera el hash segur

$sql = "INSERT INTO personal (nom, rol, username, pwdhash, host) VALUES (?, ?, ?, ?, ?)";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("sssss", $nom, $rol, $username, $pwdhash, $host);

$stmt->execute();

**4. 🔐 Login (verificació de la contrasenya)**

$username = $\_POST['username'];

$password = $\_POST['password'];

$host = $\_POST['host'];

$sql = "SELECT \* FROM personal WHERE username = ? AND host = ? AND es\_actiu = 1";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("ss", $username, $host);

$stmt->execute();

$result = $stmt->get\_result();

if ($usuari = $result->fetch\_assoc()) {

if (password\_verify($password, $usuari['pwdhash'])) {

echo "Benvingut " . $usuari['nom'];

// aquí pots crear la sessió o el que necessitis

} else {

echo "Contrasenya incorrecta.";

}

} else {

echo "Usuari o host incorrectes.";

}

**🧽 Bonus: neteja de dades antigues**

Si tenies usuaris antics amb ed25519\_password() i vols regenerar els seus pwdhash, pots fer un petit script per fer-ho un cop (si tens la contrasenya en clar, o saps regenerar-la):

$sql = "SELECT id\_personal, username FROM personal";

$result = $conn->query($sql);

while ($row = $result->fetch\_assoc()) {

$id = $row['id\_personal'];

$username = $row['username'];

$newHash = password\_hash($username, PASSWORD\_DEFAULT); // si el password és el mateix que el username

$update = $conn->prepare("UPDATE personal SET pwdhash = ? WHERE id\_personal = ?");

$update->bind\_param("si", $newHash, $id);

$update->execute();

}

Et puc ajudar a fer aquesta migració pas a pas, si vols. T'interessaria automatitzar l'actualització de tots els **pwdhash antics?**

|  |
| --- |
| **com hauria de fer un 'insert into personal values....' en DML ?** |

En DML (**Data Manipulation Language**), una instrucció INSERT INTO serveix per inserir dades dins una taula, i la forma correcta depèn de si indiques o no els noms de les columnes.

**✅ Forma Recomanada (explícita amb columnes)**

Sempre és millor indicar les columnes (especialment quan hi ha AUTO\_INCREMENT, DEFAULT, etc.):

INSERT INTO personal (nom, rol, username, pwdhash, host)

VALUES ('Kevin José CUI', 'cuiner', 'cuiner2', 'HASH\_GENERAT', 'localhost');

🔐 Substitueix 'HASH\_GENERAT' pel hash que hauries generat prèviament amb password\_hash() des de PHP.

**❌ Forma NO recomanada (omplint tots els camps)**

Aquesta només funciona si **inclous tots els camps** en el mateix ordre de definició, inclòs el id\_personal (auto\_increment), que no és recomanable fer manualment:

INSERT INTO personal

VALUES (NULL, 'Kevin José CUI', 'cuiner', 'cuiner2', 'cuiner2', 'HASH\_GENERAT', 'localhost', TRUE);

Aquesta forma no és segura perquè:

* Si canvies l'ordre o afegeixes columnes a la taula, fallarà.
* Estàs inserint la contrasenya en clar ('cuiner2'), cosa que no hauries de fer.
* Estàs tocant el id\_personal, que hauria de generar-se automàticament.

**🧑‍🍳 Exemple correcte i segur**

Si ja tens el hash generat a PHP:

$password = "cuiner2";

$pwdhash = password\_hash($password, PASSWORD\_DEFAULT);

// En SQL:

INSERT INTO personal (nom, rol, username, pwdhash, host)

VALUES ('Kevin José CUI', 'cuiner', 'cuiner2', '$pwdhash', 'localhost');

**Vols que et generi un script complet en PHP que faci aquest INSERT amb password\_hash()?**

|  |
| --- |
| **si** |

Perfecte! Aquí tens un **script complet en PHP** que fa un INSERT INTO personal utilitzant password\_hash() de forma segura, amb connexió a MariaDB via mysqli.

**✅ Script PHP: inserir un usuari nou**

<?php

// Configura la connexió a la base de dades

$host = "localhost";

$dbname = "el\_teu\_nom\_de\_base\_de\_dades";

$user = "el\_teu\_usuari";

$pass = "la\_teva\_contrasenya";

// Crear connexió

$conn = new mysqli($host, $user, $pass, $dbname);

// Comprovar connexió

if ($conn->connect\_error) {

die("Connexió fallida: " . $conn->connect\_error);

}

// Dades del nou usuari

$nom = "Kevin José CUI";

$rol = "cuiner"; // 'cuiner', 'cambrer' o 'administrador'

$username = "cuiner2";

$password = "cuiner2";

$host\_user = "localhost";

// Generar hash segur de la contrasenya

$pwdhash = password\_hash($password, PASSWORD\_DEFAULT);

// Preparar consulta

$sql = "INSERT INTO personal (nom, rol, username, pwdhash, host)

VALUES (?, ?, ?, ?, ?)";

$stmt = $conn->prepare($sql);

if (!$stmt) {

die("Error a prepare: " . $conn->error);

}

$stmt->bind\_param("sssss", $nom, $rol, $username, $pwdhash, $host\_user);

// Executar

if ($stmt->execute()) {

echo "Usuari inserit correctament.";

} else {

echo "Error a l'inserció: " . $stmt->error;

}

// Tancar connexió

$stmt->close();

$conn->close();

?>

**🔐 Consells extres**

* Pots adaptar-ho fàcilment per rebre les dades via $\_POST des d’un formulari.
* Si vols controlar duplicats (username + host), pots capturar errors de tipus duplicate key i avisar a l'usuari.

Vols també un petit formulari HTML per inserir usuaris amb aquest script?

|  |
| --- |
| **si** |

Genial! Aquí tens un petit **formulari HTML** juntament amb el mateix script PHP adaptat per rebre les dades via POST.

**📝 1. Formulari HTML (formulari\_personal.html)**

<!DOCTYPE html>

<html lang="ca">

<head>

<meta charset="UTF-8">

<title>Alta de Personal</title>

</head>

<body>

<h2>Afegir nou membre del personal</h2>

<form action="inserir\_personal.php" method="POST">

<label>Nom:</label><br>

<input type="text" name="nom" required><br><br>

<label>Rol:</label><br>

<select name="rol" required>

<option value="cuiner">Cuiner</option>

<option value="cambrer">Cambrer</option>

<option value="administrador">Administrador</option>

</select><br><br>

<label>Nom d'usuari:</label><br>

<input type="text" name="username" required><br><br>

<label>Contrasenya:</label><br>

<input type="password" name="password" required><br><br>

<label>Host:</label><br>

<input type="text" name="host" value="localhost" required><br><br>

<input type="submit" value="Afegir">

</form>

</body>

</html>

**💾 2. Script PHP (inserir\_personal.php)**

<?php

// Connexió

$host = "localhost";

$dbname = "el\_teu\_nom\_de\_base\_de\_dades";

$user = "el\_teu\_usuari";

$pass = "la\_teva\_contrasenya";

$conn = new mysqli($host, $user, $pass, $dbname);

if ($conn->connect\_error) {

die("Error de connexió: " . $conn->connect\_error);

}

// Recollida de dades

$nom = $\_POST['nom'] ?? '';

$rol = $\_POST['rol'] ?? '';

$username = $\_POST['username'] ?? '';

$password = $\_POST['password'] ?? '';

$host\_user = $\_POST['host'] ?? 'localhost';

// Validació bàsica (opcionalment pots fer més control)

if (!$nom || !$rol || !$username || !$password || !$host\_user) {

die("Falten dades.");

}

// Generar hash

$pwdhash = password\_hash($password, PASSWORD\_DEFAULT);

// Inserció

$sql = "INSERT INTO personal (nom, rol, username, pwdhash, host)

VALUES (?, ?, ?, ?, ?)";

$stmt = $conn->prepare($sql);

if (!$stmt) {

die("Error a prepare: " . $conn->error);

}

$stmt->bind\_param("sssss", $nom, $rol, $username, $pwdhash, $host\_user);

if ($stmt->execute()) {

echo "Usuari inserit correctament.";

} else {

if ($conn->errno === 1062) {

echo "Error: Ja existeix un usuari amb aquest username i host.";

} else {

echo "Error a l'inserció: " . $stmt->error;

}

}

$stmt->close();

$conn->close();

?>

**🧪 Vols que aquest formulari validi que el username no existeix abans d’enviar el formulari (via AJAX)?**

|  |
| --- |
| **si** |

Perfecte! Ara t'ensenyaré com fer una **validació en temps real amb AJAX** per comprovar si el username ja existeix **abans d’enviar el formulari**. Això millora molt l’experiència d’usuari.

**📁 Estructura dels fitxers**

projecte/

│

├── formulari\_personal.html

├── inserir\_personal.php

└── comprovar\_username.php ← (Nou fitxer per AJAX)

**📝 1. formulari\_personal.html (actualitzat amb AJAX i jQuery)**

Inclourem jQuery i un espai per missatges de validació:

<!DOCTYPE html>

<html lang="ca">

<head>

<meta charset="UTF-8">

<title>Alta de Personal</title>

<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

</head>

<body>

<h2>Afegir nou membre del personal</h2>

<form id="formulari" action="inserir\_personal.php" method="POST">

<label>Nom:</label><br>

<input type="text" name="nom" required><br><br>

<label>Rol:</label><br>

<select name="rol" required>

<option value="cuiner">Cuiner</option>

<option value="cambrer">Cambrer</option>

<option value="administrador">Administrador</option>

</select><br><br>

<label>Nom d'usuari:</label><br>

<input type="text" name="username" id="username" required><br>

<span id="missatge-username" style="color: red;"></span><br><br>

<label>Contrasenya:</label><br>

<input type="password" name="password" required><br><br>

<label>Host:</label><br>

<input type="text" name="host" value="localhost" required><br><br>

<input type="submit" value="Afegir">

</form>

<script>

$(document).ready(function() {

$('#username').on('input', function() {

const username = $(this).val();

const host = $('input[name="host"]').val();

if (username.length > 2) {

$.post('comprovar\_username.php', { username: username, host: host }, function(data) {

if (data === 'existeix') {

$('#missatge-username').text('Aquest username ja existeix!');

$('input[type="submit"]').prop('disabled', true);

} else {

$('#missatge-username').text('');

$('input[type="submit"]').prop('disabled', false);

}

});

} else {

$('#missatge-username').text('');

$('input[type="submit"]').prop('disabled', false);

}

});

});

</script>

</body>

</html>

**🆕 2. comprovar\_username.php**

Aquest fitxer repeteix part de la connexió a la base de dades i comprova si el username + host ja existeixen.

<?php

$host = "localhost";

$dbname = "el\_teu\_nom\_de\_base\_de\_dades";

$user = "el\_teu\_usuari";

$pass = "la\_teva\_contrasenya";

$conn = new mysqli($host, $user, $pass, $dbname);

if ($conn->connect\_error) {

die("Error: " . $conn->connect\_error);

}

$username = $\_POST['username'] ?? '';

$host\_user = $\_POST['host'] ?? '';

$sql = "SELECT id\_personal FROM personal WHERE username = ? AND host = ?";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("ss", $username, $host\_user);

$stmt->execute();

$stmt->store\_result();

if ($stmt->num\_rows > 0) {

echo "existeix";

} else {

echo "lliure";

}

$stmt->close();

$conn->close();

?>

**✅ Resultat**

* Quan l’usuari escriu un username, el sistema comprova si existeix i mostra un missatge.
* El botó d’enviament es bloqueja automàticament si el nom d’usuari no és vàlid.

Vols que afegim també una comprovació que la **contrasenya sigui forta (mínim caràcters, números, etc.)** abans d'enviar?

|  |
| --- |
| **si** |

Perfecte! Ara afegirem una **validació de la contrasenya en temps real al formulari**, per assegurar-nos que sigui prou forta **abans de permetre l'enviament**.

**🎯 Requisits per a la contrasenya:**

* Mínim **8 caràcters**
* Almenys **una lletra majúscula**
* Almenys **una lletra minúscula**
* Almenys **un número**

**📝 Actualització de formulari\_personal.html amb validació de contrasenya**

Afegim el control just després del camp de contrasenya:

<label>Contrasenya:</label><br>

<input type="password" name="password" id="password" required><br>

<span id="missatge-password" style="color: red;"></span><br><br>

I afegim el codi jQuery per validar la contrasenya:

<script>

function validarPassword(password) {

// Regex: mínim 8 caràcters, almenys una minúscula, una majúscula i un número

const regex = /^(?=.\*[a-z])(?=.\*[A-Z])(?=.\*\d).{8,}$/;

return regex.test(password);

}

$(document).ready(function() {

function comprovarSubmit() {

const usernameError = $('#missatge-username').text() !== '';

const password = $('#password').val();

const passwordValid = validarPassword(password);

if (usernameError || !passwordValid) {

$('input[type="submit"]').prop('disabled', true);

} else {

$('input[type="submit"]').prop('disabled', false);

}

}

$('#username').on('input', function() {

const username = $(this).val();

const host = $('input[name="host"]').val();

if (username.length > 2) {

$.post('comprovar\_username.php', { username: username, host: host }, function(data) {

if (data === 'existeix') {

$('#missatge-username').text('Aquest username ja existeix!');

} else {

$('#missatge-username').text('');

}

comprovarSubmit();

});

} else {

$('#missatge-username').text('');

comprovarSubmit();

}

});

$('#password').on('input', function() {

const password = $(this).val();

if (!validarPassword(password)) {

$('#missatge-password').text('La contrasenya ha de tenir almenys 8 caràcters, una majúscula, una minúscula i un número.');

} else {

$('#missatge-password').text('');

}

comprovarSubmit();

});

});

</script>

**✅ Resultat:**

* **El formulari només es pot enviar si:**
  + El username no existeix
  + La contrasenya compleix els requisits de seguretat
* **Missatges d’error dinàmics** avisen l’usuari abans d’enviar.

Vols que també validem que el **camp "Nom" no contingui números** o que **tots els camps estiguin plens abans de validar res més**?

|  |
| --- |
| **si** |

Perfecte! Anem a reforçar la validació del formulari:

**🎯 Noves validacions que afegirem:**

1. ✅ **El camp “Nom”** no pot contenir números ni caràcters especials.
2. ✅ Tots els camps han d’estar **plens** abans de validar el username o la contrasenya.

**📝 Codi jQuery actualitzat amb totes les validacions:**

Afegeix això dins del teu <script> al final de formulari\_personal.html:

function validarPassword(password) {

const regex = /^(?=.\*[a-z])(?=.\*[A-Z])(?=.\*\d).{8,}$/;

return regex.test(password);

}

function validarNom(nom) {

const regex = /^[A-Za-zÀ-ÿ\s\-']+$/; // Accepta lletres amb accents, espais, guions i apòstrofs

return regex.test(nom);

}

$(document).ready(function() {

function comprovarSubmit() {

const nom = $('#nom').val().trim();

const username = $('#username').val().trim();

const password = $('#password').val();

const host = $('input[name="host"]').val().trim();

const usernameError = $('#missatge-username').text() !== '';

const passwordValid = validarPassword(password);

const nomValid = validarNom(nom);

const campsPlens = nom && username && password && host;

// Missatge pel nom

if (!nomValid && nom.length > 0) {

$('#missatge-nom').text('El nom només pot contenir lletres i espais.');

} else {

$('#missatge-nom').text('');

}

// Activar o desactivar el botó

if (usernameError || !passwordValid || !nomValid || !campsPlens) {

$('input[type="submit"]').prop('disabled', true);

} else {

$('input[type="submit"]').prop('disabled', false);

}

}

$('#nom, #username, #password, input[name="host"]').on('input', function() {

comprovarSubmit();

});

$('#username').on('input', function() {

const username = $(this).val();

const host = $('input[name="host"]').val();

if (username.length > 2) {

$.post('comprovar\_username.php', { username: username, host: host }, function(data) {

if (data === 'existeix') {

$('#missatge-username').text('Aquest username ja existeix!');

} else {

$('#missatge-username').text('');

}

comprovarSubmit();

});

} else {

$('#missatge-username').text('');

comprovarSubmit();

}

});

});

**✏️ Actualitza el formulari HTML també:**

Afegim el id="nom" i el span per mostrar missatges:

<label>Nom:</label><br>

<input type="text" name="nom" id="nom" required><br>

<span id="missatge-nom" style="color: red;"></span><br><br>

**✅ Resultat:**

* **El camp Nom** no accepta números.
* **Tots els camps** han d’estar plens.
* El botó **es desactiva automàticament** si hi ha algun error.

Vols que t’ajudi a fer una versió estilitzada amb **Bootstrap 5** també?

Vols que t’ajudi a fer una versió estilitzada amb **Bootstrap 5** també?

|  |
| --- |
| **si** |

Perfecte! Aquí tens el formulari totalment estilitzat amb **Bootstrap 5**, amb totes les funcionalitats anteriors (validació en temps real, contrasenya forta, comprovació d’usuari via AJAX, etc.).

**📦 CDN de Bootstrap 5**

Afegirem Bootstrap 5 via CDN i una mica d’estil per a una aparença neta i moderna.

**📄 formulari\_personal.html amb Bootstrap 5**

<!DOCTYPE html>

<html lang="ca">

<head>

<meta charset="UTF-8">

<title>Alta de Personal</title>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">

<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

</head>

<body class="bg-light">

<div class="container mt-5">

<div class="card shadow-lg rounded-4">

<div class="card-body">

<h3 class="card-title mb-4">Afegir nou membre del personal</h3>

<form id="formulari" action="inserir\_personal.php" method="POST" novalidate>

<div class="mb-3">

<label for="nom" class="form-label">Nom complet</label>

<input type="text" class="form-control" name="nom" id="nom" required>

<div id="missatge-nom" class="form-text text-danger"></div>

</div>

<div class="mb-3">

<label for="rol" class="form-label">Rol</label>

<select class="form-select" name="rol" required>

<option value="">Selecciona un rol</option>

<option value="cuiner">Cuiner</option>

<option value="cambrer">Cambrer</option>

<option value="administrador">Administrador</option>

</select>

</div>

<div class="mb-3">

<label for="username" class="form-label">Nom d'usuari</label>

<input type="text" class="form-control" name="username" id="username" required>

<div id="missatge-username" class="form-text text-danger"></div>

</div>

<div class="mb-3">

<label for="password" class="form-label">Contrasenya</label>

<input type="password" class="form-control" name="password" id="password" required>

<div id="missatge-password" class="form-text text-danger"></div>

</div>

<div class="mb-3">

<label for="host" class="form-label">Host</label>

<input type="text" class="form-control" name="host" value="localhost" required>

</div>

<button type="submit" class="btn btn-primary w-100">Afegir</button>

</form>

</div>

</div>

</div>

<script>

function validarPassword(password) {

const regex = /^(?=.\*[a-z])(?=.\*[A-Z])(?=.\*\d).{8,}$/;

return regex.test(password);

}

function validarNom(nom) {

const regex = /^[A-Za-zÀ-ÿ\s\-']+$/;

return regex.test(nom);

}

$(document).ready(function () {

function comprovarSubmit() {

const nom = $('#nom').val().trim();

const username = $('#username').val().trim();

const password = $('#password').val();

const host = $('input[name="host"]').val().trim();

const usernameError = $('#missatge-username').text() !== '';

const passwordValid = validarPassword(password);

const nomValid = validarNom(nom);

const campsPlens = nom && username && password && host;

$('#missatge-nom').text(

nomValid || nom.length === 0 ? '' : 'El nom només pot contenir lletres i espais.');

if (!validarPassword(password) && password.length > 0) {

$('#missatge-password').text(

'Contrasenya: mínim 8 caràcters, una majúscula, una minúscula i un número.');

} else {

$('#missatge-password').text('');

}

const disable = usernameError || !passwordValid || !nomValid || !campsPlens;

$('button[type="submit"]').prop('disabled', disable);

}

$('#nom, #username, #password, input[name="host"]').on('input', function () {

comprovarSubmit();

});

$('#username').on('input', function () {

const username = $(this).val();

const host = $('input[name="host"]').val();

if (username.length > 2) {

$.post('comprovar\_username.php', {username: username, host: host}, function (data) {

if (data === 'existeix') {

$('#missatge-username').text('Aquest username ja existeix!');

} else {

$('#missatge-username').text('');

}

comprovarSubmit();

});

} else {

$('#missatge-username').text('');

comprovarSubmit();

}

});

});

</script>

</body>

</html>

**✅ Amb aquest formulari ja tens:**

* Estil modern i adaptat a dispositius.
* Validacions en temps real:
  + Nom sense números ni símbols.
  + Contrasenya segura.
  + Username únic (AJAX).
* Botó **activat/desactivat automàticament**.

Vols que també afegim un **missatge bonic de confirmació** quan s’inserta correctament? (amb modal, toast o alerta?)

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |