

Teoria de Galetes (Cookies) en PHP

1. Introducció a les Galetes

1.1. Què són les galetes?

Les **galetes** (*cookies*) són petits fitxers de text que s'emmagatzemen al navegador del client (ordinador de l'usuari) i que permeten desar informació entre diferents sol·licituds HTTP. Com que el protocol HTTP és **sense estat** (*stateless*), les galetes proporcionen un mecanisme per mantenir informació persistent entre pàgines.

1.2. Característiques principals

- **Emmagatzematge al client:** Les galetes es desen al disc dur o memòria del navegador de l'usuari
- **Petita quantitat de dades:** Normalment limitades a 4KB per galeta
- **Persistència configurable:** Poden caducar en tancar el navegador o després d'un temps determinat
- **Transmissió automàtica:** S'envien automàticament amb cada sol·licitud HTTP al mateix domini
- **Límit per domini:** Els navegadors permeten típicament unes 20-50 galetes per domini

1.3. Usos comuns de les galetes

- Recordar preferències d'usuari (idioma, tema visual)
- Mantenir sessions d'usuari (juntament amb sessions del servidor)
- Seguiment i analítiques web
- Personalització de continguts
- Carretes de compra en comerç electrònic
- Recordar credencials (amb precaució)

2. Creació de Galetes en PHP

2.1. La funció `setcookie()`

La funció `setcookie()` permet crear una galeta en PHP. **Important:** Ha de cridar-se abans de qualsevol sortida HTML.

```
setcookie(nom, valor, caducitat, ruta, domini, segur, httponly);
```

Paràmetres:

1. **nom** (obligatori): Nom de la galeta
2. **valor** (opcional): Valor a emmagatzemar (per defecte `""`)
3. **caducitat** (opcional): Timestamp UNIX de caducitat (per defecte 0 = finalitzar sessió)
4. **ruta** (opcional): Ruta del servidor on la galeta estarà disponible (per defecte `"/"`)
5. **domini** (opcional): Domini on la galeta estarà disponible
6. **segur** (opcional): Si true, només s'envia per HTTPS (per defecte false)

7. **httponly** (opcional): Si true, no accessible via JavaScript (per defecte false)

2.2. Exemples bàsics

Crear una galeta simple:

```
<?php
// Galeta que caduca quan es tanca el navegador
setcookie("nom_usuari", "Joan");
?>
```

Crear una galeta amb caducitat:

```
<?php
// Galeta que caduca en 1 hora (3600 segons)
$caducitat = time() + 3600;
setcookie("nom_usuari", "Joan", $caducitat);
?>
```

Crear una galeta amb més opcions:

```
<?php
$caducitat = time() + 3600; // 1 hora
setcookie("nom_usuari", "Joan", $caducitat, "/", "", false, true);
?>
```

2.3. Càlcul de la caducitat

La caducitat s'especifica com a **timestamp UNIX** (segons des de l'1 de gener de 1970).

```
<?php
// Temps actual
time()

// Sumar temps
time() + 3600           // 1 hora
time() + (60 * 60)      // 1 hora
time() + 86400          // 1 dia (24 hores)
time() + (7 * 24 * 60 * 60) // 1 setmana
time() + (30 * 24 * 60 * 60) // 1 mes (aproximat)
?>
```

3. Lectura de Galetes en PHP

3.1. L'array superglobal \$_COOKIE

Totes les galetes enviades pel navegador estan disponibles a l'array superglobal \$_COOKIE.

```
<?php
// Comprovar si existeix una galeta
if (isset($_COOKIE['nom_usuari'])) {
    $nom = $_COOKIE['nom_usuari'];
    echo "Hola, " . $nom;
} else {
    echo "No hi ha galeta";
}
?>
```

3.2. Bon ús de isset()

Sempre cal comprovar si una galeta existeix abans d'accedir-hi per evitar errors:

```
<?php
// CORRECTE
if (isset($_COOKIE['nom'])) {
    echo $_COOKIE['nom'];
}

// INCORRECTE - pot generar error si no existeix
echo $_COOKIE['nom'];
?>
```

3.3. Accés segur amb validació

```
<?php
function obtenir_galeta($nom, $per_defecte = "") {
    if (isset($_COOKIE[$nom])) {
        return $_COOKIE[$nom];
    }
    return $per_defecte;
}

$nom_usuari = obtenir_galeta('nom_usuari', 'Anònim');
echo "Hola, " . $nom_usuari;
?>
```

4. Modificació de Galetes

Per modificar una galeta, simplement cal tornar a cridar `setcookie()` amb el mateix nom però amb un valor diferent:

```
<?php
// Crear galeta inicial
setcookie("comptador", "1", time() + 3600);

// Més tard, modificar-la
if (isset($_COOKIE['comptador'])) {
    $nou_valor = $_COOKIE['comptador'] + 1;
    setcookie("comptador", $nou_valor, time() + 3600);
}
?>
```

Important: El canvi no es veurà reflectit a `$_COOKIE` fins a la següent càrrega de pàgina.

5. Esborrament de Galetes

5.1. Establir caducitat al passat

Per esborrar una galeta, cal establir la seva caducitat a un moment del passat:

```
<?php
// Esborrar galeta
setcookie("nom_usuari", "", time() - 3600);
?>
```

5.2. Funció per a esborrar galetes

```
<?php
function esborrar_galeta($nom) {
    setcookie($nom, "", time() - 3600, "/");
}

esborrar_galeta('nom_usuari');
?>
```

5.3. Esborrament de múltiples galetes

```
<?php
function esborrar_totes_galetes() {
    foreach ($_COOKIE as $nom => $valor) {
        setcookie($nom, "", time() - 3600, "/");
    }
}

esborrar_totes_galetes();
?>
```

6. Consideracions Importants

6.1. Headers Already Sent

L'error més comú amb galetes és "Headers already sent". Això passa quan s'intenta cridar `setcookie()` després d'haver enviat contingut HTML.

INCORRECTE:

```
<html>
<body>
<?php
setcookie("nom", "valor"); // ERROR!
?>
```

CORRECTE:

```
<?php
setcookie("nom", "valor");
?>
<html>
<body>
```

6.2. La galeta no està disponible immediatament

Quan es crea una galeta amb `setcookie()`, no estarà disponible a `$_COOKIE` fins a la següent sol·licitud:

```
<?php
setcookie("prova", "valor");

// Això serà false
if (isset($_COOKIE['prova'])) {
    echo "Existeix"; // No s'executarà
}

// Cal recarregar la pàgina
```

?>

6.3. Seguretat

Validació i sanitització:

```
<?php
// Mai confiar directament en les dades de galetes
if (isset($_COOKIE['edat'])) {
    $edat = filter_var($_COOKIE['edat'], FILTER_VALIDATE_INT);
    if ($edat !== false && $edat > 0) {
        echo "Edat: " . $edat;
    }
}
?>
```

Escapar sortides:

```
<?php
if (isset($_COOKIE['nom'])) {
    // Escapar per prevenir XSS
    echo htmlspecialchars($_COOKIE['nom'], ENT_QUOTES, 'UTF-8');
}
?>
```

6.4. Limitacions

- Mida màxima: ~4KB per galeta
- Nombre màxim: 20-50 galetes per domini (depenent del navegador)
- Les galetes poden ser desactivades pel navegador
- Les galetes poden ser modificades pel client
- No emmagatzemar informació sensible (contrasenyes, dades bancàries)

7. Exemple Complet: Recorda Nom d'Usuari

```
<?php
// Temps de caducitat: 1 hora
$temps_caducitat = 3600;

// Processa formulari si s'ha enviat
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    if (isset($_POST['nom']) && isset($_POST['cognom'])) {
        $nom = trim($_POST['nom']);
        $cognom = trim($_POST['cognom']);

        // Creació de galetes
        setcookie('nom', $nom, time() + $temps_caducitat, '/');
        setcookie('cognom', $cognom, time() + $temps_caducitat, '/');

        // Redirecció per actualitzar
        header('Location: ' . $_SERVER['PHP_SELF']);
        exit;
    }
}

// Processament d'esborrament de galetes
if (isset($_GET['esborrar'])) {
    setcookie('nom', '', time() - 3600, '/');
    setcookie('cognom', '', time() - 3600, '/');
```

```

// Redirecció
header('Location: ' . $_SERVER['PHP_SELF']);
exit;
}

// Comprova si existeixen galetes
$te_galetes = isset($_COOKIE['nom']) && isset($_COOKIE['cognom']);
?>
<!DOCTYPE html>
<html lang="ca">
<head>
    <meta charset="UTF-8">
    <title>Exemple Galetes</title>
</head>
<body>
    <?php if ($te_galetes): ?>
        <h1>Benvingut, <?php echo htmlspecialchars($_COOKIE['nom'] . ' ' .
$_COOKIE['cognom']); ?></h1>
        <form method="get">
            <button type="submit" name="esborrar" value="1">Esborrar
galetes</button>
        </form>
    <?php else: ?>
        <h1>Primer cop que ens visites</h1>
        <form method="post">
            <label>Nom: <input type="text" name="nom" required></label><br>
            <label>Cognom: <input type="text" name="cognom"
required></label><br>
            <button type="submit">Desa</button>
        </form>
    <?php endif; ?>
</body>
</html>

```

8. Bones Pràctiques

1. **Sempre validar i sanititzar** les dades de galetes abans d'usar-les
2. **Emprar httponly** per prevenir accés via JavaScript (contra XSS)
3. **Emprar secure** per connexions HTTPS
4. **Establir caducitat adequada** segons les necessitats
5. **No emmagatzemar informació sensible** a les galetes
6. **Comprovar sempre amb isset()** abans d'accedir a una galeta
7. **Cridar setcookie()** abans de qualsevol sortida HTML
8. **Especificar la ruta** per evitar problemes amb subdirectoris
9. **Donar noms descriptius** a les galetes
10. **Documentar la finalitat** de cada galeta al codi

9. Depuració de Galetes

9.1. Veure totes les galetes

```

<?php
echo "<pre>";
print_r($_COOKIE);
echo "</pre>";
?>

```

9.2. Comprovació de caducitat

Les galetes no inclouen informació sobre la seva caducitat en PHP. Cal gestionar-ho manualment si es necessita:

```
<?php
// Desa timestamp de creació
setcookie('timestamp_creacio', time(), time() + 3600);

// Més tard, comprova
if (isset($_COOKIE['timestamp_creacio'])) {
    $temps_transcorregut = time() - $_COOKIE['timestamp_creacio'];
    echo "Han passat " . $temps_transcorregut . " segons";
}
?>
```

10. Resum

Les galetes són un mecanisme fonamental per mantenir estat entre sol·licituds HTTP. Permeten emmagatzemar petites quantitats d'informació al navegador del client i són essencials per a funcionalitats com recordar preferències, mantenir sessions i personalitzar l'experiència de l'usuari. És crucial usar-les amb cura, sempre validant les dades, establint caducitats adequades i tenint en compte les consideracions de seguretat.