Teoria de Sessions en PHP

1. Introducció a les Sessions

1.1. Què són les sessions?

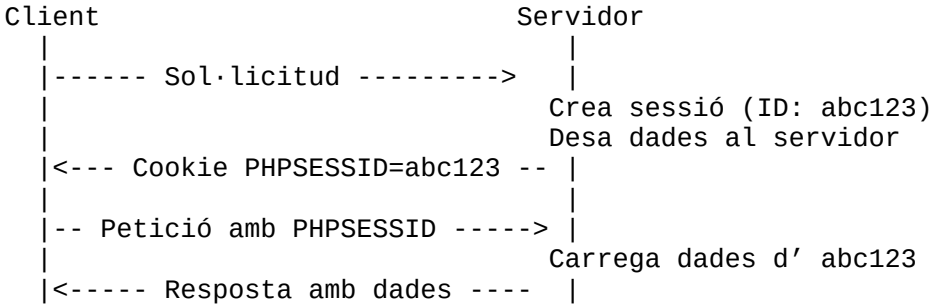
Les **sessions** són un mecanisme per a emmagatzemar informació de l'usuari al servidor (no al client) durant la seva visita a un lloc web. Permeten mantenir l'estat entre diferents pàgines i sol·licituds HTTP.

1.2. Diferència entre galetes i sessions

Característica	Galetes (Cookies)	Sessions
Emmagatzematge	Client (navegador)	Servidor
Mida	~4KB	Il·limitada (dins de límits del servidor)
Seguretat	Menys segur	Més segur
Caducitat	Configurable	Finalitza en tancar navegador o *timeout*
Velocitat	Més ràpid (local)	Més lent (servidor)
Dades sensibles	NO recomanat	Recomanat

1.3. Com funcionen les sessions?

1. **L'usuari accedeix** al lloc web
2. El servidor **crea una sessió única** i genera un ID de sessió (PHPSESSID)
3. **L'ID es tramet al client** mitjançant una galeta
4. Les **dades es desen al servidor** associades a aquest ID
5. A **cada petició**, el client envia l'ID de sessió
6. El servidor **recupera les dades** associades a aquesta ID



1.4. Usos comuns de les sessions

- Autenticació d'usuaris (*login*/*logout*)
- Carros de compra
- Preferències temporals de l'usuari
- Dades de formularis de diversos passos
- Control d'accés a pàgines
- Prevenció de CSRF (Cross-Site Request Forgery)

2. Gestió Bàsica de Sessions

2.1. Iniciar una sessió - session_start()

SEMPRE cal iniciar la sessió abans d'usar-la. Ha de cridar-se abans de qualsevol sortida HTML.

```
<?php
session_start();
?>
```

Important:

- Només cal cridar-la **una vegada per pàgina**
- Ha d'estar **abans de qualsevol HTML**
- Cal cridar-la a **totes les pàgines** que utilitzin sessions

2.2. Emmagatzemar dades - $_SESSION

L'array superglobal `$_SESSION` permet desar i recuperar dades de sessió.

```
<?php
session_start();

// Desar dades
$_SESSION['nom'] = 'Joan';
$_SESSION['edat'] = 25;
$_SESSION['regals'] = array('Llibre', 'Pilota', 'Rellotge');
?>
```

2.3. Llegir dades de sessió

```
<?php
session_start();

// Comprovar si existeix
if (isset($_SESSION['nom'])) {
    echo "Hola, " . $_SESSION['nom'];
} else {
    echo "No heu iniciat sessió";
}
?>
```

2.4. Modificar dades de sessió

```
<?php
session_start();

// Modifica simplement reassignant
$_SESSION['nom'] = 'Maria';

// Modifica array
$_SESSION['regals'][] = 'Joc de taula';  // Afegeix
unset($_SESSION['regals'][0]);           // Elimina element
?>
```

2.5. Elimina variables individuals

```
<?php
session_start();

// Elimina una variable específica
unset($_SESSION['nom']);

// Les altres variables de sessió continuen existint
?>
```

3. Finalització de Sessions

3.1. Destrucció de totes les dades de sessió

```
<?php
session_start();

// Esborra totes les variables de sessió
$_SESSION = array();

// Destrueix la sessió
session_destroy();
?>
```

3.2. Funció completa de *logout*

Per fer un *logout* complet i segur:

```
<?php
session_start();

// 1. Esborra totes les variables
$_SESSION = array();

// 2. Elimina la galeta de sessió
if (isset($_COOKIE[session_name()])) {
    setcookie(session_name(), '', time() - 3600, '/');
}

// 3. Destrueix la sessió
session_destroy();

// 4. Redirecció
header('Location: index.php');
exit;
?>
```

3.3. Diferència entre unset() i session_destroy()

```
<?php
session_start();

// OPCIÓ 1: Esborra UNA variable
unset($_SESSION['nom']);
// Les altres variables continuen existint

// OPCIÓ 2: Esborra TOTES les variables però manté la sessió
$_SESSION = array();
// La sessió existeix però sense dades

// OPCIÓ 3: Destrueix completament la sessió
session_destroy();
// Cal session_start() per crear-ne una de nova
?>
```

4. Configuració de Sessions

4.1. Nom de la sessió

```
<?php
// Canvia el nom de la sessió (abans de session_start)
session_name('MEU_SITE');
session_start();
?>
```

4.2. Temps de caducitat (*timeout*)

Configuració a php.ini:

```
session.gc_maxlifetime = 3600  ; 1 hora en segons
```

Configuració en temps d'execució:

```
<?php
// Estableix timeout de 30 minuts
ini_set('session.gc_maxlifetime', 1800);
session_start();
?>
```

4.3. *Timeout* personalitzat amb *timestamp*

```
<?php
session_start();

$temps_maxim = 1800; // 30 minuts en segons

// Primera visita: desa timestamp
if (!isset($_SESSION['ultim_acces'])) {
    $_SESSION['ultim_acces'] = time();
}

// Comprova si ha caducat
if (time() - $_SESSION['ultim_acces'] > $temps_maxim) {
    // Sessió caducada
    session_unset();
    session_destroy();
```

```
    header('Location: login.php?timeout=1');
    exit;
}

// Actualitza timestamp
$_SESSION['ultim_acces'] = time();
?>
```

4.4. Directori d'emmagatzematge

```
<?php
// Canvia directori a on es desen les sessions
session_save_path('/ruta/al/directori');
session_start();
?>
```

5. Seguretat en Sessions

5.1. Regeneració de la ID de sessió

Per a prevenir atacs de fixació de sessió (*session fixation*):

```
<?php
session_start();

// Després de fer login
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Validar credencials...

    // Regenera ID de sessió
    session_regenerate_id(true);

    $_SESSION['usuari'] = $usuari;
}
?>
```

5.2. Validació de sessió

```
<?php
session_start();

// Comprova si l'usuari està autenticat
function esta_autenticat() {
    return isset($_SESSION['usuari_id']) && !empty($_SESSION['usuari_id']);
}

// Protegeix pàgina
if (!esta_autenticat()) {
    header('Location: login.php');
    exit;
}
?>
```

5.3. Vinculació de sessió a IP (opcional)

Atenció: Pot causar problemes amb IPs dinàmiques.

```
<?php
session_start();

// En fer login
if (!isset($_SESSION['ip'])) {
    $_SESSION['ip'] = $_SERVER['REMOTE_ADDR'];
}

// Comprovar IP
if ($_SESSION['ip'] !== $_SERVER['REMOTE_ADDR']) {
    // Possible atac
    session_unset();
    session_destroy();
    header('Location: login.php?error=security');
    exit;
}
?>
```

5.4. Escapament de sortides

```
<?php
session_start();

// SEMPRE escapeu dades de sessió abans de mostrar-les
if (isset($_SESSION['nom'])) {
    echo htmlspecialchars($_SESSION['nom'], ENT_QUOTES, 'UTF-8');
}
?>
```

6. Exemple Complet: Sistema de Regals

```
<?php
// inici.php - Seleccioneu regals i nom
session_start();

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Desa nom i cognom
    $_SESSION['nom'] = trim($_POST['nom']);
    $_SESSION['cognom'] = trim($_POST['cognom']);

    // Desa regals seleccionats
    $_SESSION['regals'] = array();
    if (isset($_POST['regals']) && is_array($_POST['regals'])) {
        $_SESSION['regals'] = $_POST['regals'];
    }

    // Redirecció al cistell
    header('Location: cistell.php');
    exit;
}
?>
<!DOCTYPE html>
<html lang="ca">
<head>
    <meta charset="UTF-8">
    <title>Selecció de Regals</title>
</head>
```

```
<body>
    <h1>Esculliu els vostres regals</h1>
    <form method="post">
        <label>Nom: <input type="text" name="nom" required></label><br>
        <label>Cognom: <input type="text" name="cognom" required></label><br>

        <h3>Regals disponibles:</h3>
        <label><input type="checkbox" name="regals[]" value="Llibre">
Llibre</label><br>
        <label><input type="checkbox" name="regals[]" value="Pilota">
Pilota</label><br>
        <label><input type="checkbox" name="regals[]" value="Rellotge">
Rellotge</label><br>
        <label><input type="checkbox" name="regals[]" value="Joc"> Joc de
taula</label><br>

        <button type="submit">Continueu</button>
    </form>
</body>
</html>

<?php
// cistell.php - Mostra i actualitza cistell
session_start();

// Comprovació que hi ha dades
if (!isset($_SESSION['nom']) || !isset($_SESSION['cognom'])) {
    header('Location: inici.php');
    exit;
}

// Actualitza regals si s'ha tramès el formulari
if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['actualitzar'])) {
    $_SESSION['regals'] = array();
    if (isset($_POST['regals']) && is_array($_POST['regals'])) {
        $_SESSION['regals'] = $_POST['regals'];
    }
}

// Finalització de sessió
if (isset($_GET['finalitzar'])) {
    session_unset();
    session_destroy();
    header('Location: inici.php');
    exit;
}

$nom_complet = htmlspecialchars($_SESSION['nom'] . ' ' . $_SESSION['cognom']);
$regals = isset($_SESSION['regals']) ? $_SESSION['regals'] : array();
?>
<!DOCTYPE html>
<html lang="ca">
<head>
    <meta charset="UTF-8">
    <title>Cistell de Regals</title>
</head>
<body>
    <h1>Cistell de <?php echo $nom_complet; ?></h1>

    <h2>Regals seleccionats:</h2>
    <?php if (empty($regals)): ?>
        <p>No heu seleccionat cap regal</p>
    <?php else: ?>
```

```
        <ul>
            <?php foreach ($regals as $regal): ?>
                <li><?php echo htmlspecialchars($regal); ?></li>
            <?php endforeach; ?>
        </ul>
    <?php endif; ?>

    <h3>Actualitzeu regals:</h3>
    <form method="post">
        <?php
        $opcions = array('Llibre', 'Pilota', 'Rellotge', 'Joc de taula');
        foreach ($opcions as $opcio):
            $checked = in_array($opcio, $regals) ? 'checked' : '';
        ?>
            <label>
                <input type="checkbox" name="regals[]" value="<?php echo $opcio;
?>" <?php echo $checked; ?>>
                <?php echo $opcio; ?>
            </label><br>
        <?php endforeach; ?>

        <button type="submit" name="actualitzar">Actualitza</button>
    </form>

    <hr>
    <a href="?finalitzar=1">Finalitza sessió</a>
</body>
</html>
```

7. Gestió d'Errors Comuns

7.1. "Headers already sent"

Error:

```
Warning: session_start(): Cannot send session cookie - headers already sent
```

Causa: HTML abans de `session_start()`

Solució:

```
<?php
// CORRECTE: session_start() PRIMER
session_start();
?>
<html>
```

7.2. *Session* no es manté entre pàgines

Causes possibles:

1. No cridar `session_start()` a cada pàgina
2. Les galetes estan desactivades
3. Problemes amb el directori de sessions

Solució:

```
<?php
// A TOTES les pàgines que usen sessions
session_start();
?>
```

7.3. Dades de sessió no s'actualitzen

Problema:

```
<?php
session_start();
$_SESSION['comptador'] = $_SESSION['comptador'] + 1;
// No veu el canvi immediatament
?>
```

Solució: Les dades estan disponibles immediatament, però recordeu desar-les:

```
<?php
session_start();

if (!isset($_SESSION['comptador'])) {
    $_SESSION['comptador'] = 0;
}

$_SESSION['comptador']++;
// Ja està disponible
echo $_SESSION['comptador'];
?>
```

8. Funcions Útils de Sessions

```
<?php
// Obtenir ID de sessió actual
$sessio_id = session_id();

// Obtenir nom de la sessió
$nom_sessio = session_name();

// Comprovar si hi ha una sessió activa
if (session_status() === PHP_SESSION_ACTIVE) {
    echo "Sessió activa";
}

// Veure totes les variables de sessió
print_r($_SESSION);
?>
```

9. Bones Pràctiques

1. **Sempre crideu session_start()** a l'inici de cada pàgina que usi sessions
2. **Regenereu ID després de login** per a prevenir fixació de sessió
3. **Establiu *timeout* adient** segons les necessitats
4. **Valideu dades de sessió** abans d'usar-les
5. **Escapeu sortides** amb htmlspecialchars()
6. **Destruiu sessió completament** en fer logout
7. **No emmagatzemeu contrasenyes** a la sessió
8. **Empreu HTTPS** per a prevenir interceptació
9. **Comproveu autenticació** a pàgines protegides
10. **Netegeu sessions antigues** periòdicament al servidor

10. Depuració

10.1. Veure contingut de sessió

```
<?php
session_start();

echo "<h3>Contingut de \$_SESSION:</h3>";
echo "<pre>";
print_r($_SESSION);
echo "</pre>";
?>
```

10.2. Informació de configuració

```
<?php
echo "<h3>Configuració de sessions:</h3>";
echo "Nom: " . session_name() . "<br>";
echo "ID: " . session_id() . "<br>";
echo "Directori: " . session_save_path() . "<br>";
echo "GC Maxlifetime: " . ini_get('session.gc_maxlifetime') . " segons<br>";
?>
```

11. Resum

Les sessions són fonamentals per mantenir estat al servidor en aplicacions web. Permeten emmagatzemar dades de l'usuari de forma segura i persistent durant la seva visita. És essencial cridar `session_start()` a cada pàgina, validar les dades, regenerar IDs després de canvis d'autenticació i destruir sessions adequadament. Les sessions són més segures que les galetes per emmagatzemar informació sensible, però requereixen gestió correcta per evitar problemes de seguretat.