



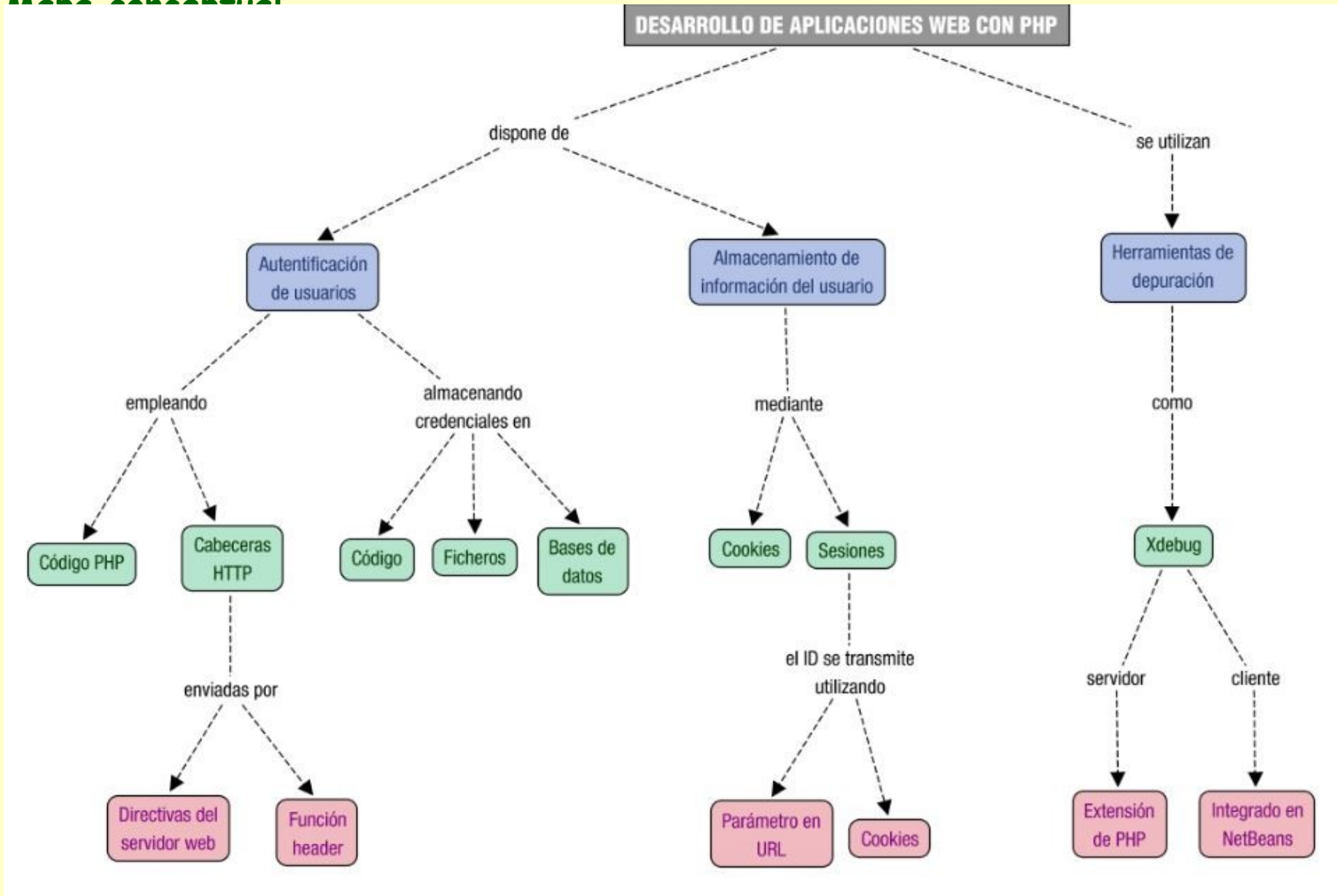
Tema 5: Desarrollo de aplicaciones web utilizando código embebido



- Mantenimiento del estado.
- Sesiones.
- Cookies. Aplicaciones prácticas. Las cookies y el navegador.
- Seguridad: usuarios, perfiles, roles.
- Autentificación de usuarios. Mecanismos.
- Herramientas de programación.
- Pruebas y depuración.

Tema 5: Desarrollo de aplicaciones web utilizando código embebido

Mapa conceptual





Tema 5: Desarrollo de aplicaciones web utilizando código embebido



Autenticación de usuarios y control de acceso

Identificar al servidor certificado del servidor (HTTPS)

Identificar al usuario - Control de acceso

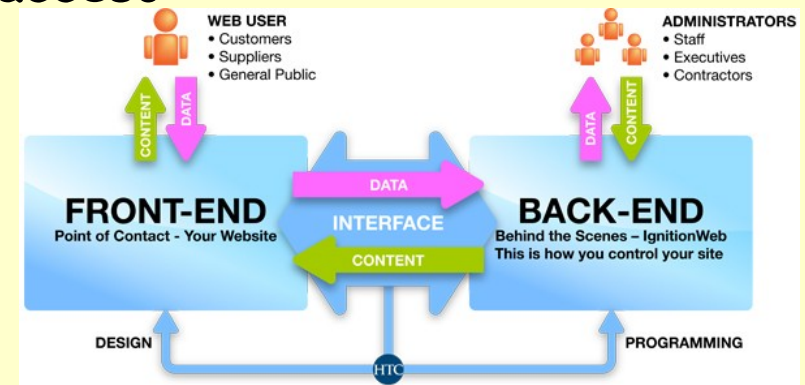
Mecanismos de autenticación

Certificados digitales de los usuarios - DNI electrónico

Usuario / Password (utilizando HTTPS para enviarlos de forma segura)

Front End - Back End

Tipos de usuarios - Perfiles - Nivel de acceso





Tema 5: Desarrollo de aplicaciones web utilizando código embebido



Autenticación de usuarios y control de acceso

Configuración de la solicitud de usuario/password

- (I) **.htaccess** Fichero que define los recursos con acceso restringido
AuthType Basic / Digest

- (II) **header()** Función de control de acceso
 validando el acceso en el código (sin base de datos)
 validando el acceso consultando a la base de datos

Autenticación HTTP con PHP



Tema 5: Desarrollo de aplicaciones web utilizando código embebido



Autenticación de usuarios y control de acceso

Mecanismo de autenticación (I)

`/etc/apache2/users`

Fichero de usuarios HTTP

`htpasswd [-c] users usuario`

Crea el usuario y lo añade al fichero users

`.htaccess` Fichero que define los recursos con acceso restringido (**AuthType**, **AuthUserFile**)

Fichero de configuración del apache, incluir **AllowOverride** para que utilice `.htaccess`

Habilitar .htaccess



Tema 5: Desarrollo de aplicaciones web utilizando código embebido



Autenticación de usuarios y control de acceso

Mecanismo de autenticación (II)

`$_SERVER`

Variable superglobal

`$_SERVER['PHP_AUTH_USER']`

Nombre de usuario introducido

`$_SERVER['PHP_AUTH_PW']`

Contraseña introducida

`$_SERVER['PHP_AUTH_TYPE']`

Método de autenticación HTTP (Basic o Digest)

`header()`

Función que obliga al servidor a solicitar usuario / password

No es necesario usar el fichero `.htaccess`

Tabla Usuarios en base de datos
cifrado (MD5, SHA, ...)

Almacena el usuario y el password

Autenticación HTTP con PHP



Tema 5: Desarrollo de aplicaciones web utilizando código embebido



Autenticación de usuarios y control de acceso

Tabla Usuario

CodUsuario

DescUsuario

Password

Perfil - TipoUsuario - CodPerfil - NivelDeAcceso

Algoritmos de cifrado de password

SHA1, SHA2

MD5

...

Autenticación HTTP con PHP



Tema 5: Desarrollo de aplicaciones web utilizando código embebido



Cookies

cookie fichero de texto que el navegador guarda en el entorno del cliente

setcookie ('nombreDeLaCookie', \$variableConElValor, fechaDeCaducidad) si no hay fecha de caducidad la cookie se borra cuando se cierra el navegador

\$_COOKIE['nombreDeLaCookie'] Una variable tipo array asociativo de variables pasadas al script actual a través de Cookies HTTP.

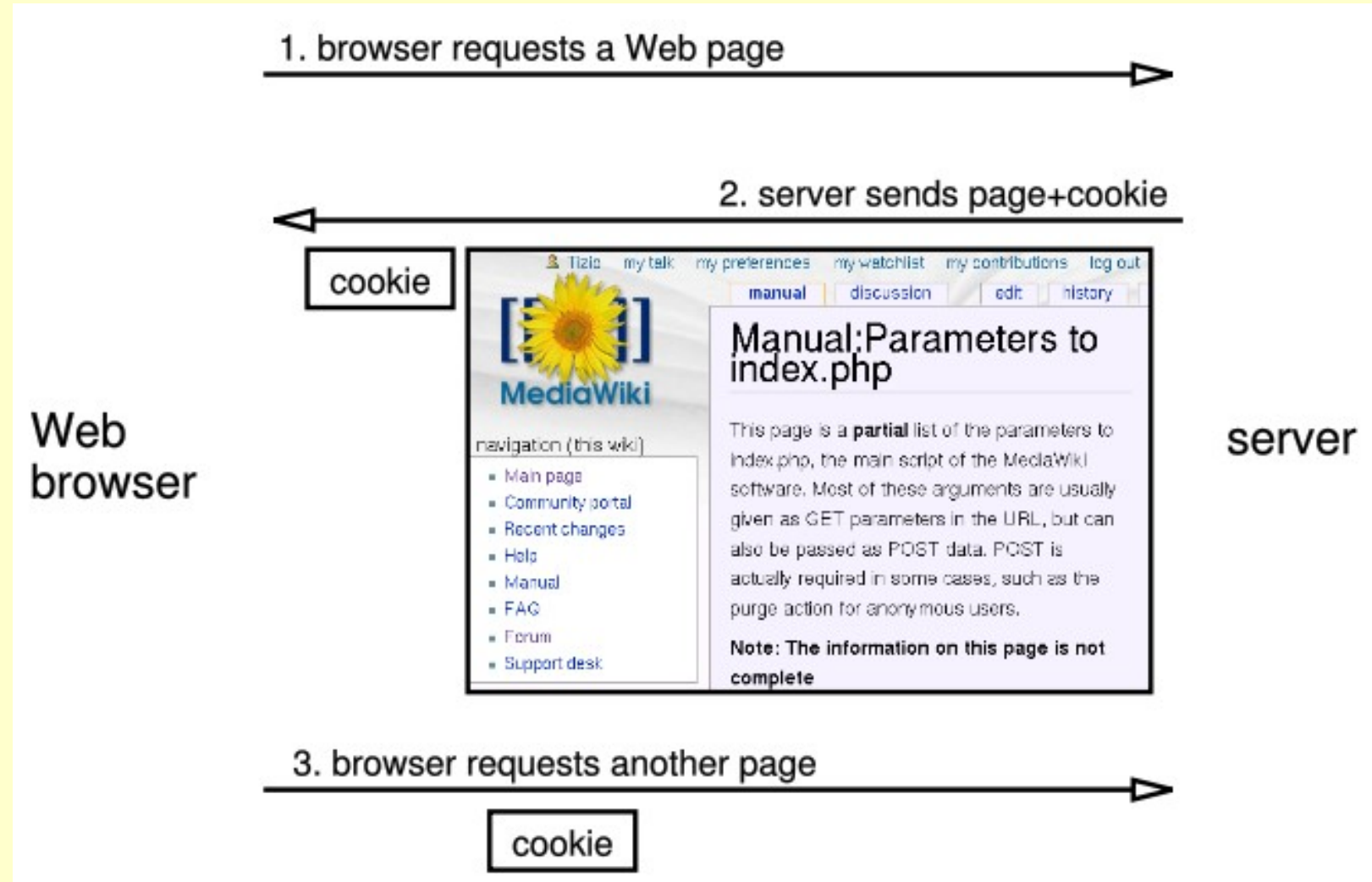
Las sentencias setcookie() deben enviarse antes de que el navegador muestre nada en la pantalla.

El navegador envía de forma automática el contenido de las cookies al servidor web, donde podremos recuperarlas en \$_COOKIE

cookies

Tema 5: Desarrollo de aplicaciones web utilizando código embebido

Cookies



cookies



Tema 5: Desarrollo de aplicaciones web utilizando código embebido



Manejo de sesiones

sesión: El término sesión hace referencia al conjunto de información relativa a un usuario concreto.

SID Identificador de sesión (**usuario**)

Server Side Cookies - Cookies del lado del servidor : información (de la sesión) del usuario almacenada en el servidor.

Opciones de identificación y propagación del SID

- Usando las **cookies**
- Propagando el SID en la **URL&PHPSESSID=identificadorDeSesion**

Si utilizas la URL para propagar el SID, este se perderá cuando cierres tu navegador. Sin embargo si utilizas cookies el SID se mantendrá mientras no se borre la cookie.

Ver configuración de de las directivas de sesiones en `phpinfo()`

Funciones de sesión



Tema 5: Desarrollo de aplicaciones web utilizando código embebido



Manejo de sesiones

<code>\$_SESSION</code>	Variable superglobal de sesión
<code>session_id()</code>	SID

Inicio y fin de sesión

`session_auto_start()` si está activada la directiva `session.auto_start` en la configuración PHP, la sesión comienza automáticamente cuando se conecta el usuario

`session_start()` fuerza el inicio de sesión o **reanuda la anterior**

`session_unset()` elimina las variables almacenadas en la sesión actual

`session_destroy()` elimina las variables almacenadas de la sesión actual y del dispositivo de almacenamiento utilizado

Funciones de sesión



Tema 5: Desarrollo de aplicaciones web utilizando código embebido



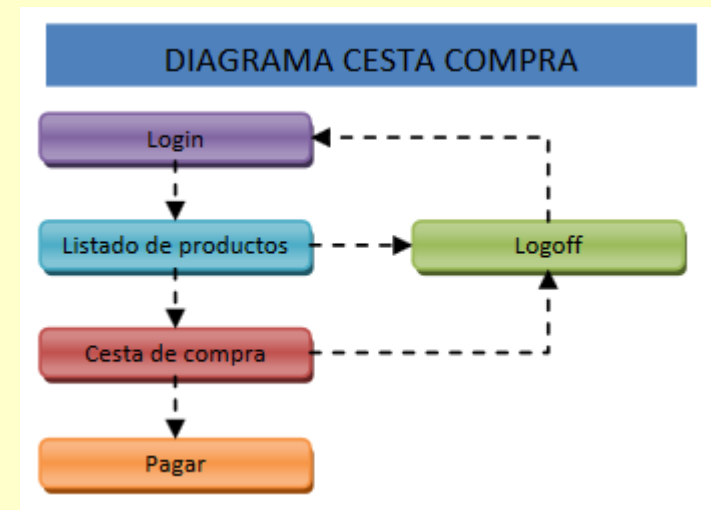
Gestión de la información de la sesión

Manejo de las sesiones

Manejo de los errores

Funcionalidad

Comunicación entre páginas



Funciones de sesión



Tema 5: Desarrollo de aplicaciones web utilizando código embebido



Herramientas para depuración de código

Herramientas básicas:

`echo()`
`print()`
`print_r()`
`var_dump()`

Herramientas específica de depuración:

Xdebug

nos permite depurar sin necesidad de modificar el código
integrada en NetBeans

no es necesario que Xdebug esté en la misma máquina que el
cliente NetBeans

Xdebug estará en la misma máquina que el servidor web php

Xdebug



Tema 5: Desarrollo de aplicaciones web utilizando código embebido



Instalación, configuración y administración de Xdebug en Linux

Instalación

```
apt-get install php-pear  
apt-get install php5-dev  
apt-get install xdebug
```

Activar en php.ini
[Xdebug]

Reiniciar apache
/etc/init.d/apache2 restart



(*) También podemos instalar Xdebug en Windows

Xdebug



Tema 5: Desarrollo de aplicaciones web utilizando código embebido



Depuración de código con Xdebug

Creación de registros con las llamadas a funciones que se produzcan, incluyendo parámetros y valores devueltos.

Creación de registros de optimización que permiten analizar el rendimiento de los programas.

Depuración remota.

Empleo de funciones sobre NetBeans:

- Ejecución paso a paso

- Punto de parada / Continuar con la ejecución

- Inspección de variables

- Modificación del valor de las variables

Xdebug



Tema 5: Desarrollo de aplicaciones web utilizando código embebido



Funciones interesantes

hash() //Genera un valor cifrado de un string utilizando distintos algoritmos de función resumen (SHA, MD5,...)

md5() //Resumen MD5

sha1() //Resumen SHA1

hashfile() //Calcula el resumen de un fichero

crypt() //Hash de cadenas de un solo sentido

Extensiones criptográficas

¿Preguntas?

