GT Xunta OOo

Dirección Xeral de Educación, Formación Profesional e Innovación Educativa

Material para a formación profesional inicial

Familia profesional	IFC	Informática e comunicacións
Ciclo formativo	CMIFC01	Sistemas microinformáticos e redes
Grao		Medio
Módulo profesional	MP0228	Aplicacións web
Unidade didáctica	UD01	Características das aplicacións web
Actividade	**A03**	**Funcionamento das aplicacións web.**
Autores		
Nome do arquivo		CMIFC01_MP0228_UD01_A03_Funcionamento_apps_web

© 2010 Xunta de Galicia.

Consellería de Educación e Ordenación Universitaria.

Aviso Legal: Reservados todos os dereitos.

A presente obra está protexida pola lei, que establece penas de prisión e/ou multas, ademais das correspondentes indemnizacións por danos e prexuízos, para quen reproduza, plaxie ou distribúa o seu contido en calquera tipo de soporte sen a preceptiva autorización.

Índice# Ficha técnica

Nº	Título	Descrición	Duración
A03	Funcionamento das aplicacións web	Descrición dos compoñentes principais dun servidor de aplicacións web. Utilización de plataformas integradas para proba e desenrolo de aplicacións web.	4

Resultados de aprendizaxe do currículo

Resultados de aprendizaxe do currículo	Completo
* RA1. Instala xestores de contidos configurados consonte certos requisitos, e identifica as súas aplicacións.	Non

Obxectivos didácticos e título e descrición da actividade

	Obxectivos específicos	Actividade		Descrición básica	Duración
O3.1	Identificar as tecnoloxías e linguaxes empregados polas aplicacións web en xeral e os sistemas xestores de contidos en particular.	A3	Funcionamento das aplicacións web.	Descrición dos compoñentes principais dun servidor de aplicacións web. Utilización de plataformas integradas para proba e desenrolo de aplicacións web.	4
O3.2	Identificar os requisitos xerais para a instalación de aplicacións web.				

Criterios de avaliación

Criterios de avaliación

* CA1.2.1. Identificáronse os requisitos xerais para instalar xestores de contidos.

- Identificáronse as tecnoloxías e linguaxes imbricados na execución de aplicacións web.
- Instaláronse e probáronse plataformas integradas orientadas á proba e desenvolvemento de aplicacións web.

Contidos

Contidos

* Requisitos para a instalación de xestores de contidos (CA1.2.1).

- Funcionamento dos xestores de contidos.
- Emprego de plataformas integradas para a proba e desenvolvemento de aplicacións web.

Actividades de ensino e aprendizaxe e de avaliación, métodos, recursos e instrumentos de avaliación

Qué e para qué	Cómo		Con qué	Cómo e con qué se valora	Duració
Actividade(título e descrición)	**Profesorado(en termos de tarefas)**	**Alumnado(tarefas)**	**Resultados ou produtos**	**Recursos**	**Instrumentos e procedementos de avaliación**	
A3. Funcionamento das aplicacións web • Descrición dos compoñentes principais dun servidor de aplicacións web. • Utilización de plataformas integradas para proba e desenrolo de aplicacións web.	* **Tp3.1** Descrición por parte do profesor do modo xeral de funcionamento dunha aplicación web. Identificación dos conceptos de servidor web e sistema xestor de bases de datos. • **Tp3.2** Descrición polo profesor das linguaxes empregadas na execución de aplicacións web. Execución de código no servidor e no cliente.	* **Ta3.1** Descarga e instalación guiada por parte dos alumnos de plataformas integradas para a proba e desenvolvemento de aplicacións web. Configuración e verificación do seu funcionamento. • **Ta3.2** Realización polos alumnos dunha actividade guiada de xeración dun documento web ?Ola mundo? a partires dun código PHP sinxelo, identificando	* Instalación de XAMPP nos ordenadores dos alumnos. • Configuración, descarga e instalación de MoWeS portable e Bitnami nos ordenadores dos alumnos. • Código PHP ?Ola mundo? e linguaxe HTML resultante.	* Ordenadores dos alumnos con sistemas operativos Linux e Windows. • www.apachefriends.org • www.chsoftware.net/en/mowes/mowesportable/main.htm • bitnami.org	* Diario de clase. • Lista de cotexamento relativa á instalación, configuración e probas realizadas ás plataformas web. • Cuestionario sobre os contidos e tarefas da actividade.	

Qué e para qué	Cómo	Con qué	Cómo e con qué se valora	Duració
	Linguaxes do servidor (PHP, ASP, JSP) e do cliente (HTML, XHTML, CSS, Javascript). • **Tp3.3** Exposición por parte do profesor da interrelación existente entre as distintas tecnoloxías implicadas na execución de aplicacións web. • **Tp3.4** Exposición polo profesor das plataformas integradas para a proba e desenvolvemento de aplicacións web. Identificación das plataformas máis empregadas e dos seus compoñentes. Plataformas portables	as fases de execución. Verificación do código de marcas obtido polo navegador web.		

1. **A2. Funcionamento das aplicacións web**
    1. **Introdución**

Na actividade que nos ocupa aprenderanse os seguintes conceptos e manexo de destrezas:

- Os compoñentes precisos nun servidor para a execución de aplicacións web e o modo de interoperación entre os mesmos.
- O procedemento que executa o servidor para obter a linguaxe de marcado a partires do código dunha aplicación web.
- As linguaxes máis comúns que se empregan na actualidade para programar aplicacións web.
- A instalación e utilización de plataformas integradas orientadas á proba e desenvolvemento de aplicacións web.

1. 1. **Actividade**

Xeración de páxinas webs dinámicas

O funcionamento das aplicacións web basease no mesmo procedemento que se emprega para que un navegador web amose unha páxina web de Internet.

O principio amósase na figura seguinte. Un navegador inicia unha comunicación (1) cun servidor web a través de Internet. Nesa comunicación o navegador indícalle ao servidor a páxina que quere obter. O servidor busca esa páxina (2), e se a ten almacenada a envía (3) ao navegador. Cando a páxina chega ao navegador, este interpreta o seu contido e a amosa na pantalla.

[[Image:]]

Este principio é o mesmo que se aplica a calquera comunicación entre un cliente e un servidor. Chamámoslle de forma xenérica cliente ao extremo que inicia a comunicación e solicita a información. No outro lado atópase sempre o servidor, que é unha máquina que espera ás conexións do cliente e resposta enviando a información que se lle solicita.

No caso das comunicacións web, o cliente web será un navegador que se comunicará cun servidor web.

Este esquema sinxelo foi superado xa hai tempo, de tal forma que o servidor web xeralmente non se limita a enviar páxinas que ten gardadas. Na web actual podemos diferenciar entre:

- Páxinas estáticas. Son aquelas que se rexen polo principio anterior. Atópanse almacenadas no servidor preparadas para ser enviadas sen modificación aos clientes que as soliciten.
- Páxinas dinámicas. Como o seu nome indica, estas páxinas xenáranse no servidor cando se solicitan de tal forma que o seu contido pode variar en función de certos parámetros (información enviada dende o cliente, páxinas enviadas anteriormente, dirección IP dende a que se solicita, ...).

Compoñentes dun servidor de aplicacións web

Para enviarlle a un navegador páxinas web estáticas, chega cun servidor web. Sen embargo, a xeración dinámica de páxinas web farase seguindo as instrucións contidas nun programa, que deberá executarse no servidor (aínda que non necesariamente na mesma máquina).

Ademais, a xeración dinámica de páxinas web normalmente vai asociada á utilización das mesmas para manexar información. Isto trae consigo a necesidade de almacenar esa información de forma adecuada, para o que se soe facer uso de bases de datos.

Por tanto, para xerar de forma dinámica páxinas web, deberemos empregar no noso servidor os seguintes compoñentes:

- Un servidor web para recibir as peticións do cliente e enviarlle a páxina unha vez xerada. O servidor deberá distinguir cando unha solicitude se corresponde a unha páxina estática (páxinas htm e html), e cando a unha dinámica (páxinas con outras extensións como por exemplo php). Neste caso deberá reenviarlle a solicitude ao módulo encargado da súa xeración.
- O módulo necesario para a obtención da páxina web a partir do programa na que se define a mesma. Deberá integrase co servidor web, e dependerá da linguaxe de programación empregada. As linguaxes máis habituais son PHP, ASP / ASP.Net e Java.
- Unha aplicación de base de datos que normalmente será tamén un servidor (isto é, execútase como un programa que escoita peticións de datos e as resposta). O servidor máis empregado na actualidade e MySQL, aínda que existen outras alternativas como PostgreSQL.

- É moi importante non confundir as linguaxes empregadas na xeración dinámica de páxinas web con aquelas que se executan no navegador web. No TA1 acláranse estes conceptos.

O esquema de funcionamento xenérico dun servidor web é o seguinte.

[[Image:]]

- O navegador ou cliente web envíalle (1) unha petición a un servidor web.

- O servidor busca (2) a páxina solicitada entre as que ten almacenadas.
- No caso de que se trate dunha páxina que debe xerarse de forma dinámica (por exemplo, cando a extensión é .php, .asp, .jsp ou .aspx), envíase (3) ao módulo de execución.
- No proceso de execución pode ser necesario comunicarse (4) cun servidor de bases de datos, por exemplo para obter información que se incluirá na páxina de saída.
- Cando a páxina xa foi xerada (5), o servidor envíalla (6) ao cliente do mesmo xeito que se fai coas páxinas estáticas.

Por exemplo, imos ver o que sucede cando empregamos o navegador para ver o noso correo (empregando gmail, hotmail ou calquera outro servidor de correo web).

- Abrimos o noso navegador e imos á páxina de inicio do servizo de correo web (na que temos que introducir o noso usuario e contrasinal). Esta pode ser unha páxina web estática (o seu contido non cambia) e nese caso o servidor a colle e a envía.
- Cando enchemos a páxina co noso usuario e contrasinal, o noso navegador os envía ao servidor web. O máis habitual é que o servidor responda amosando a nosa bandexa de entrada coas mensaxes da nosa conta. Esta páxina é un claro exemplo dunha páxina web dinámica.

[[Image:]]

- Obviamente o navegador non envía a mesma páxina a cada usuario. Cando un usuario inicia sesión co seu usuario e contrasinal, o servidor web executa un programa que obtén dunha base de datos a información da nosa conta (mensaxes, contactos, etc) e a emprega para xerar a páxina que nos chega.

Aplicacións web

As aplicacións web empregan a xeración dinámica de páxinas web no servidor para crear aplicacións que se executen nun servidor web e se amosen nun navegador. Estas aplicacións teñen certas vantaxes e inconvintes se as comparamos coas aplicacións tradicionais que se executan no sistema operativo da propia máquina.

Vantaxes:

- Non é necesario instalalas nos equipos nos que se van a empregar. Instálanse e execútanse soamente nun equipo, no servidor, e isto é suficiente para que se poidan empregar de forma simultánea dende moitos.
- Como soamente se atopan instaladas nun equipo, é moi sinxelo xestionalas (facer copias de seguridade dos datos, corrixir erros, actualizalas).
- Pódense empregar en todos aqueles sistemas que teñan un navegador web, independentemente das súas características (non é necesario un equipo potente) e do seu sistema operativo.
- Pódense empregar dende calquera sitio no que se teña unha conexión co servidor. En moitos casos isto fai posible que se poida acceder ás mesmas desde sistemas non convencionais, como por exemplo teléfonos móbiles.

Inconvintes.

- O interface de usuario das mesmas é a páxina que se amosa no navegador. Isto restrinxe as características do mesmo a aquelas dunha páxina web.
- Dependemos dunha conexión co servidor para poder empregalas. Se nos falla a conexión, non podermos acceder ás aplicacións web.
- A información que se amosa no cliente debe transmitirse dende o servidor ao cliente. Isto fai que certo tipo de aplicacións non sexan adecuadas para a súa implementación como aplicación web (por exemplo, as aplicacións que manexan arquivos multimedia).

Plataformas integradas para a proba e desenvolvemento de aplicacións web

Como acabamos de ver, para executar unha aplicación web necesitamos instalar os seguintes compoñentes no noso servidor: un servidor web, un módulo para a execución dos programas, e un servidor de bases de datos.

Cando imos preparar unha máquina como servidor de aplicacións en Internet, estes pasos son moi importantes e convén facelos con coidado. Pero se o que queremos é poñer en funcionamento un servidor de aplicacións privado para probar ou configurar aplicacións web, podemos empregar un paquete AMP.

Chámanse de xeito xenérico AMP a paquetes software que inclúen os elementos necesarios para a execución de aplicacións web. Estes paquetes veñen listos para ser instalados en poucos pasos de xeito que non é necesario levar a cabo tarefas de configuración posteriores.

[[Image:]]

O nome AMP é un acrónimo que se corresponde ao nome dos seus compoñentes principais: o servidor web Apache, a servidor de bases de datos MySQL, e a linguaxe de programación PHP. Ademais, falamos de paquetes WAMP cando se instalan e executan en contornos Windows, e paquetes LAMP cando veñen preparados para sistemas operativos Linux.

Entre o software AMP que podemos empregar, imos ver tres opcións que se diferencian no seu método de instalación e utilización.

BitNami

[[Image:]]

BitNami é a máis sinxela se queremos probar unha aplicación web. Atópase dispoñible para plataformas Linux (LAMP) e Windows (WAMP). Para cada unha das principais aplicacións web, podemos elixir entre:

- Descargar unha aplicación que instala a plataforma (WAMP ou LAMP) e a aplicación web de forma conxunta sobre o noso sistema operativo.
- Descargar e instalar a plataforma (WAMP ou LAMP) adecuada ao noso sistema operativo, e engadirlle posteriormente o módulo da aplicación web que desexemos (un ou varios).
- Descargar unha máquina virtual con Linux coa aplicación web xa instalada e lista para ser executada.

Se nos decidimos por unha das dúas primeira opcións, debemos ter en conta que os instaladores para Linux (LAMP) veñen en formato executable con extensión .bin, polo que tras baixalos é preciso marcalos como executables (chmod +x) e executalos preferiblemente como superusuario empregando sudo (en caso contrario teríamos algúns problemas, como non poder usar a ruta /opt ou que o servidor Apache non poida empregar o porto 80 para as comunicacións e teña que configurarse no 8080).

Además, no proceso de instalación dásenos a opción de engadir a aplicación web PhpMyAdmin para xestionar vía web a base de datos MySQL, e no caso de instalar tamén algunha aplicación web teremos que engadir a conta de administración asociada á mesma.

Tamén teremos que indicar o nome da máquina no que estamos a instalar a aplicación, que se empregará nas comunicacións en rede. Se soamente imos a empregar o BitNami dende a nosa propia máquina e non precisamos darlle acceso dende Internet ou dende unha rede de área local, chega con indicar como nome da máquina ?localhost?.

Se queremos optar pola terceira opción, máis sinxela, podemos escoller as máquinas virtuais de BitNami con sistema operativo Ubuntu ou openSuSE. Ambas veñen preparadas para executarse co software de VMWare. Podémolas executar coa aplicación gratuíta VMWare Player (http://www.vmware.com/products/player/).

Se por exemplo queremos probar Gallery (aplicación web para xestionar álbums de fotos), dentro do sitio web de BitNami podemos ir á páxina web de Gallery (http://bitnami.org/stack/gallery).

[[Image:]]

E das opcións que temos:

[[Image:]]

Escollemos descargar a máquina virtual de Ubuntu.

Unha vez descargada, a descomprimimos e executamos dende o VMWare Player. É importante ter en conta que as máquinas virtuais de BitNami xa están preparadas para executarse, e entre os parámetros de configuración cos que veñen figuran o nome e o contrasinal dos usuarios.

Cando iniciamos a máquina virtual, amosará unha pantalla na que se indica a dirección de acceso á aplicación web.

Nestes intres a aplicación web xa está executándose. Non é necesario entrar no sistema operativo da máquina virtual, pero se o necesitamos deberemos facelo co usuario ?bitnami? (contrasinal ?bitnami?), tal e como se indica.

[[Image:]]

Se empregamos un navegador para conectarnos á dirección que se indica (http://192.168.169.129 no noso exemplo), veremos unha páxina especial de BitNami.

[[Image:]]

Premendo na icona accedemos á aplicación web instalada (Gallery).

[[Image:]]

Xa temos a nosa aplicación web instalada e lista para que a empreguemos.

Neste caso podemos ver na web que o nome de usuario da aplicación web é ?user? e o seu contrasinal é ?bitnami?. É preferible cambiar estes valores no noso primeiro acceso.

[[Image:]]

- Realice a tarefa 1 correspondente á descarga, instalación e proba dun paquete BitNami sobre un sistema operativo Ubuntu previamente preparado.

MoWeS

[[Image:]]

MoWeS, que soamente está dispoñible para contornos Windows (WAMP), ten dúas características diferenciadoras sobre a opción anterior:

- Permite xerar e descargar un paquete de instalación persoalizado que inclúa unha ou varias aplicacións web.
- Unha vez descargado instálase de forma portable, isto é, podemos executar a aplicación web dende unha memoria USB e levala a outros equipos informáticos.

Para crear o noso paquete con MoWeS, abrimos a páxina web http://www.chsoftware.net/en/mowes/mowesmixer/mowesmixer.htm, e escollemos a opción de obter un novo paquete.

[[Image:]]

Na seguinte páxina decidimos que compoñentes figurarán no noso paquete. A páxina divídese en dous grupos. No primeiro escollemos principalmente as versións do servidor web, a base de datos e a linguaxe de programación. No segundo grupo marcamos as aplicacións web que se incluirán.

[[Image:]]

Nos escolleremos as versións actuais de Apache2, MySQL5 e PHP5, así como a aplicación web Gallery2.

Para evitar que o sistema se empregue de forma mecanizada, deberemos introducir a resposta a unha sinxela pregunta.

[[Image:]]

E a continuación xenárase o paquete e comeza a descarga do mesmo.

[[Image:]]

Unha vez descomprimido, comezamos a instalación executando mowes.exe. Escollemos a linguaxe entre as dispoñibles...

[[Image:]]

... e observamos como se van instalando os paquetes que escollemos anteriormente.

[[Image:]]

Se non temos problemas, poderemos ver a pantalla da aplicación cos servizos iniciados.

[[Image:]]

E abrise tamén unha páxina web no noso navegador.

[[Image:]]

Avísanos de que nas aplicacións web instaladas, o usuario será ?admin? e o contrasinal ?password?. Se pulsamos en ?Back?, dinos a lista de paquetes instalados e para cada unha das aplicacións web temos un enlace directo.

[[Image:]]

Cando accedemos á aplicación web vemos unha páxina semellante á que obtivemos con BitNami (neste caso inclúese tamén un álbum de proba).

[[Image:]]

XAMPP

[[Image:]]

XAMPP é máis flexible que as dúas anteriores, pero a cambio non é tan doado empregala para poñer en funcionamento aplicacións web.

Está dispoñible para sistemas Windows e Linux, pero a súa instalación non inclúe ningunha aplicación web. Limítase a instalar os requisitos dos que falamos anteriormente: o servidor web Apache, o servidor de bases de datos MySQL e a linguaxe PHP. Unha vez instalado XAMPP, para poñer en funcionamento unha aplicación web deberemos descargala e instalala seguindo as súas instrucións particulares.

Tamén temos a opción de facer unha instalación portable de XAMPP, tal e como sucedía con MoWeS.

XAMPP descárgase dende a dirección http://www.apachefriends.org/es/xampp.html. Seleccionamos a versión para o noso sistema operativo. Nos faremos a instalación en Windows.

[[Image:]]

A versión principal de XAMPP inclúe moitas aplicacións que non empregaremos nesta actividade. De feito, dende a mesma páxina imos descargar unha versión reducida (?Lite?) de XAMPP. Podemos descargar calquera das dúas opcións (a EXE ou a ZIP). Descargamos a EXE (ocupa moito menos) e a executamos para descomprimila nunha carpeta do noso sistema.

[[Image:]]

Unha vez descomprimida, execútase un asistente que realiza a configuración de XAMPP. Primeiro pregunta se queremos engadir accesos directos.

[[Image:]]

Comproba se o directorio no que se atopa XAMPP é o mesmo que figura na súa configuración. A primeira vez que se instala teremos que contestar que corrixa a configuración (y). Se queremos instalar XAMPP de xeito portable, teremos que repetir este paso se ao movelo dun sistema a outro se cambia, por exemplo, a letra da unidade da memoria USB.

Se fora o caso, para executar este paso ?a man? temos que ir á carpeta de XAMPP (xampplite), e executar setup_xampp.bat.

[[Image:]]

Despois pregunta se queremos instalar XAMPP de xeito portable. Se escollemos facelo así (y), non se instalarán servizos no noso equipo e deberemos iniciar o servidor web (Apache) e o servidor de base de datos (MySQL) de xeito manual cando os precisemos.

[[Image:]]

Coa información que lle facilitamos, o asistente prepara XAMPP para que o poidamos empregar.

[[Image:]]

Ao finalizar o asistente, preséntanos un menú dende o que podemos iniciar o panel de control de XAMPP (opción 1; este paso tamén o podemos realizar dende os accesos directos que se crearon no noso sistema no primeiro paso do asistente).

[[Image:]]

Iniciamos o panel de control de XAMPP e pechamos o asistente coa opción ?x?.

No panel de control teremos que iniciar os servizos de Apache e MySQL.

[[Image:]]

Tras iniciar os servizos, xa podemos acceder a XAMPP empregando o navegador web (http://localhost).

Na primeira execución pídenos que escollamos a linguaxe do interface.

[[Image:]]

A pantalla inicial de XAMPP ten a seguinte aparencia.

[[Image:]]

Como vemos, non se inclúe ningunha aplicación web pero sí algunhas utilidades para, por exemplo, comprobar o correcto funcionamento do sistema.

Se sobre este XAMPP quixéramos instalar algunha aplicación web, teríamos que facelo agora seguindo as instrucións concretas da mesma.

- Realice a tarefa 2 correspondente á descarga e instalación dun paquete XAMPP nun sistema operativo Ubuntu previamente preparado.

Proba de execución de código no servidor

Empregando o XAMPP Lite que rematamos de instalar, imos comprobar o seu funcionamento ao procesar páxinas web dinámicas. Imos crear un código sinxelo na linguaxe PHP e almacenalo no servidor (o XAMPP) para despois solicitarllo dende o cliente (o navegador).

Das carpetas que se crearon no noso sistema ao instalar XAMPP, os documentos que empregará o servidor web almacénanse na carpeta htdocs. Se listamos o seu contido nun XAMPP recen instalado,

[[Image:]]

observamos que non está baleira. Os arquivos que almacena correspóndense á pantalla inicial do propio XAMPP.

Para poder diferenciar as aplicacións do servidor, cada unha é mellor almacenala na súa propia carpeta. Por tanto, o primeiro que faremos será crear unha carpeta ?proba? xusto baixo htdocs para gardar nela o programa que imos a facer.

De seguido, cun editor de textos (por exemplo o Bloc de Notas de Windows), tecleamos o seguinte código en PHP e gardamos o arquivo na carpeta proba co nome ?proba.php?.

[[Image:]]

A páxina que fixemos encárgase de amosar dúas liñas nunha páxina web. A primeira liña é un texto fixo ?Hola mundo!? e unha etiqueta HTML de salto de liña (<br>). A segunda liña coll

Para probar o seu funcionamento, abrimos no navegador a dirección http://localhost/proba/proba.php.

[[Image:]]

Se amosamos o código fonte da páxina, que é o que realmente chega ao navegador dende o servidor web, podemos comprobar a diferenza entre a páxina que nos xeramos e o que realmente chega ao cliente.

[[Image:]]

1. 1. **Tarefas**

As tarefas propostas son as seguintes.

- Tarefa 1. **Descarga, instalación e proba dun paquete BitNami sobre un sistema operativo Linux previamente instalado**. Nesta tarefa, partindo dun sistema operativo Ubuntu Desktop instalado, descargaremos e instalaremos un paquete BitNami xa preparado coa aplicación web Joomla!.

- Tarefa 2. **Descarga e instalación dun paquete XAMPP nun sistema operativo Linux previamente instalado**. Nesta tarefa faremos o mesmo que fixemos con XAMPP no desenrolo da actividade, empregando nesta ocasión un Ubuntu Desktop.

1. 1. 1. **Tarefa 1. Descarga, instalación e proba dun paquete BitNami**

Para a realización desta tarefa debemos dispoñer dun sistema operativo Linux (pode ser Ubuntu Desktop aínda que debería funcionar con calquera das principais distribucións), xa instalado. Descargaremos e instalaremos un paquete BitNami xa preparado coa aplicación web Joomla!.

Autoavaliación

Imos á web de BitNami, e na páxina de Gallery (http://bitnami.org/stack/gallery), dende a sección ?Native? descargamos a versión para Linux.

[[Image:]]

Unha vez descargada, a executamos para instalala no sistema operativo. Para iso, temos que abrir un terminal e dende o mesmo, na carpeta ?Descargas?, primeiro debemos marcar o arquivo como executable e logo executalo como superusuario.

[[Image:]]

Comezará entón o proceso de instalación. No mesmo temos que indicar se además de Gallery queremos ou non instalar tamén PhpMyAdmin (non é preciso nesta tarefa).

[[Image:]]

Despois de indicar a carpeta de instalación (podemos deixar a que ven por defecto), teremos que introducir os datos para a conta de usuario para a aplicación web Gallery.

[[Image:]]

E o nome da máquina en que estamos a facer a instalación. Se como é o caso soamente a imos a empregar de forma local, podemos poñer como nome da máquina ?localhost?.

[[Image:]]

Cando remata a instalación podemos acceder á mesma dende Firefox na máquina virtual, coa dirección http://localhost.

[[Image:]]

Se queremos iniciar sesión en Gallery, accedemos a través do enlace e introducimos os datos que nos foron solicitados na instalación.

[[Image:]]

1. 1. 1. **Tarefa 2. Descarga e instalación dun paquete XAMPP en Linux**

Para a realización desta tarefa debemos dispoñer dun sistema operativo Linux (pode ser Ubuntu Desktop aínda que debería funcionar con calquera das principais distribucións), xa instalado.

Descargaremos un paquete XAMPP (do mesmo xeito que fixemos anteriormente con XAMPP Lite empregando Windows) e o instalaremos. Podemos seguir as instrucións da páxina http://www.apachefriends.org/en/xampp-linux.html.

Autoavaliación

Dende a web de XAMPP, entramos na páxina correspondente á versión para Linux.

[[Image:]]

Descargamos o arquivo (XAMPP para Linux non ten versión Lite), que é un tar comprimido con gzip. Para descomprimilo, dende un terminal facemos.

[[Image:]]

Xa temos XAMPP instalado. Para inicialo facemos.

[[Image:]]

Podemos entrar na dirección http://localhost/ para comprobar o seu funcionamento.

1. **Materiais**
    1. **Textos de apoio ou de referencia**

- **Core Web Programming, 2nd Ed**, Marty Hall e Larry Brown, Ed. Prentice Hall.
- **Web Server Programming**, Neil Gray, Ed. John Wiley & Sons.
- **TA1**. Linguaxes de cliente e de servidor.

1. 1. **Recursos didácticos**

- Ordeador persoal, preferentemente con software de virtualización.
- Sistemas operativos Windows e Linux.
- Software BitNami, MoWeS e XAMPP tal e como se indica na actividade e nas tarefas correspondentes.

1. **Avaliación**

Criterios de avaliación seleccionados para esta actividade	Instrumento de avaliación
* CA1.2.1. Identificáronse os requisitos xerais para instalar xestores de contidos.	* Lista de cotexamento relativa á instalación, configuración e probas realizadas ás plataformas web. • Cuestionario.
* Identificáronse as tecnoloxías e linguaxes imbricados na execución de aplicacións web.	* Cuestionario.
* Instaláronse e probáronse plataformas integradas orientadas á proba e desenvolvemento de aplicacións web.	* Diario de clase.

Cuestionario sobre os requisitos xerais para instalar xestores de contidos e a identificación de tecnoloxías e linguaxes imbricados na execución de aplicacións web

Exemplo de cuestionario:

1. Cál dos seguintes compoñentes non é necesario nun servidor de aplicacións web?

- Linguaxe de programación.
- Servidor web.

- Servidor ftp.

1. A qué fai referencia a letra P no acrónimo AMP?

- Á palabra ?programming?.

- Á linguaxe de programación PHP.

- Á linguaxe de programación Perl.

1. Cál das seguintes solucións pode descargarse en forma de máquina virtual xa configurada?

- BitNami.

- MoWeS.
- XAMPP.

1. Con cál das seguintes solucións se pode xerar un paquete de instalación persoalizado?

- BitNami.

- MoWeS.

- XAMPP.

1. Cál das seguintes solucións non inclúe paquetes de instalación con aplicación web incorporada?

- BitNami.
- MoWeS.

- XAMPP.

1. Cál dos seguintes non é un servidor de base de datos?

- MySQL.

- Apache.

- PostgreSQL.

1. Cál das seguintes linguaxes non procesa o navegador web?

- PHP.

- HTML.
- Javascript.

1. É imprescindible unha base de datos na execución de aplicacións web?

- Non.

- Sí.
- Depende da linguaxe empregada.

1. Cómo distingue o servidor web se unha páxina contén código que debe ser procesado antes de enviala?

- Examina a páxina para ver se existe o código.
- Dependendo da carpeta na que se atope a páxina.

- Dependendo da extensión da páxina.

1. Qué ocorre cando abrimos cun navegador unha páxina que contén código PHP sen empregar un servidor web?

- O navegador non comprende o código PHP que contén a páxina.

- O navegador busca un servidor web para que procese o código da páxina.
- O navegador emprega o módulo da linguaxe PHP para executar o código.

Lista de cotexamento relativa á instalación, configuración e probas realizadas ás plataformas web

- Diferenciáronse os distintos requerimentos de instalación das opcións dispoñibles parar a plataforma BitNami.
- Descargouse a plataforma adecuada ao sistema operativo empregado.
- Levouse a cabo correctamente a instalación e configuración do servidor de aplicacións.

- Empregouse a URL correcta para abrir a aplicación web dende o navegador.
- Identificouse a diferenza entre os usuarios do sistema operativo e os usuarios das aplicacións web.

1. **TA1. Linguaxes de cliente e de servidor**

Cando falamos de programación web, podemos diferenciar dous tipos de linguaxes. Aqueles que se executan no servidor (neste caso falamos de programación do lado do servidor), e aqueles que se executan no cliente (programación do lado do cliente).

Os primeiros son os que serven para que o servidor poida xerar de forma dinámica as páxinas que envía aos navegadores. Nunca se envían ao navegador. O que envía o servidor é o resultado da execución dos programas feitos nestas linguaxes. Pódense empregar con este fin moitas linguaxes e tecnoloxías diferentes.

Os segundos forman parte da páxina que chega ao navegador dende o servidor web. Non se executan no servidor senón no navegador web. Aportan funcionalidades ás páxinas para, por exemplo, poder comprobar os datos que se introducen nos formularios web ou crear efectos ou animacións. A linguaxe máis empregada na programación do lado cliente é Javascript.

Linguaxe de marcas HTML

As páxinas web que o servidor envía ao navegador compóñense de etiquetas da linguaxe HTML (HyperText Markup Language). É doado entender o básico da linguaxe HTML aprendendo algunhas etiquetas para poder crear páxinas sinxelas, e por outra parte as últimas versións ofrecen características para poder facer páxinas web con deseños avanzados.

As etiquetas dunha páxina web empréganse para marcar elementos e permite definir a estrutura do seu contido, indicando por exemplo, qué parte é o encabezado, cáles son os parágrafos, as partes que forman as táboas, etc. Diferéncianse do resto do documento ao rodearse dos símbolos < e >. Por exemplo,
 é unha etiqueta que permite inserir liñas baleiras nunha páxina web, e <p> define cada un dos parágrafos do texto.

Por exemplo, o código HTML seguinte.

[[Image:]]

Cando o procesa o navegador amósase.

[[Image:]]

Información e estilo dunha páxina web

Aínda que o obxectivo inicial de HTML era definir soamente a estrutura do contido da páxina, pronto comezaron a aparecer etiquetas que aportaban información relacionada co estilo dos elementos, como <u> para suliñar, <b> para resaltar ou <font> para indicar un tipo de letra específico.

Para separar a información da estrutura dun documento da correspondente ao seu estilo, creouse a linguaxe CSS. Os documentos CSS compóñense dunha serie de regras de estilo; cada unha das regras define un estilo e pódese aplicar a un ou a varios elementos dunha páxina web.

Por exemplo, se introducimos unha folla de estilos CSS na anterior páxina web.

[[Image:]]

E creamos a folla de estilos seguinte, que indica que o texto contido en todas as etiquetas <p> se amose en cor verde.

[[Image:]]

Cando a páxina se amose no navegador, veremos o seguinte.

[[Image:]]

Linguaxes de programación de lado cliente

A linguaxe Javascript permítenos que as páxinas web empreguen o navegador para realizar outro tipo de tarefas que non sexan solicitar e amosar páxinas web.

A maioría de aplicacións web empregan Javascript dun xeito ou doutro, e nalgunhas resulta unha ferramenta imprescindible para a realización de determinadas tarefas.

Como exemplo, imos engadir un pequeno código Javascript á nosa páxina web para que obteña e amose o tipo de navegador que estamos a empregar.

[[Image:]]

Cando o navegador obtén e procesa a páxina web, amosa unha ventá coa seguinte información.

[[Image:]]

Se observamos o código da páxina que chega ao navegador, vemos que todo o código da mesma é enviado polo servidor web.

[[Image:]]

Linguaxes de programación de lado servidor

Se cando falamos de programación no lado cliente (no navegador) falamos case que soamente de Javascript, no lado servidor temos varias opcións. A máis popular é a linguaxe PHP.

Imos a cambiar a páxina web anterior para que, empregando PHP, obteña e amose a ubicación na que o servidor almacena as páxinas web. É moi importante ter en conta que neste caso a extensión da páxina debe ser .php, para que o servidor saiba que contén código que debe ser executado antes de enviala.

[[Image:]]

No navegador obtemos o seguinte.

[[Image:]]

E como podemos comprobar, o código que introducimos na páxina non chega ao navegador web; soamente o resultado da súa execución.

[[Image:]]