

IMPLANTACIÓN DE ARQUITECTURAS WEB

Distintos modelos de arquitecturas WEB

Índice

1.	Internet: Conceptos básicos	3
1.1	A Web.....	3
2.	Aplicacións web	3
2.1	Vantaxes.....	4
2.2	Inconvenientes	4
2.3	Modelos de arquitectura	4
	Modelo cliente/servidor	5
	Modelo orientado a mensaxes.....	5
	Modelo Orientado a servizos	5
	Modelo orientado a recursos	5
2.4	Estrutura	5
2.5	Software que intervéñ	7
3.	Protocolo HTTP	7
3.1	Mensaxes de petición	8
	Petición GET.....	8
	Petición POST	8
3.2	Mensaxes de resposta.....	8
	Códigos de estado	9
3.3	Tipos MIME	9

1. Internet: Conceptos básicos

Internet é un sistema global de redes de computadoras interconectadas entre si que utilizan a familia de protocolos TCP/IP para compartir recursos e intercambiar información entre miles de millóns de dispositivos en todo o mundo. Trátase dunha rede de redes descentralizada que, grazas a dita familia de protocolos, funciona como unha única rede a pesares da heteroxeneidade das redes físicas que a compoñen.

Ten a súa orixe na rede ARPANET (Advanced Research Projects Agency Network), creada por encargo do Departamento de Defensa dos EEUU en 1969, coa finalidade de comunicar organismos oficiais, de xeito que, se houbera unha catástrofe que destruíse algún deles, o resto da rede continuase funcionando.

Algúns dos servizos de Internet aínda amplamente empregados hoxe en día son:

- **Correo electrónico.** Servizo de correspondencia que permite enviar ou recibir mensaxes textuais e outros documentos desde un emisor a un ou varios destinatarios.
- **FTP (File Transfer Protocol).** Servizo que permite a transferencia de ficheiros entre máquinas.
- **WWW (World Wide Web).** Este servizo é un sistema de documentos de hipertexto (texto enriquecido con ligazóns) interconectados entre si. É a ferramenta máis utilizada de Internet e permite aos usuarios acceder, mediante un navegador, a páxinas web que poden conter texto, imaxes, vídeos e outro contido multimedia e navegar entre eles mediante hiperligazóns.

1.1 A Web

A World Wide Web foi creada en 1989 por Tim Berners-Lee mentres traballaba no CERN (Organización Europea para a Investigación Nuclear). Consiste nun sistema de organizar a información mediante páxinas conectadas a través de hiperligazóns, que utiliza para a súa transferencia a Internet como medio físico e o protocolo HTTP (Hypertext Transfer Protocol), para que os navegadores web poidan facer as peticións de ditas páxinas aos servidores web e recibir as respostas sobre as mesmas. Defínese tamén a linguaxe HTML (HyperText Markup Language) para a edición de páxinas web.



Nos primeiros 3 minutos e medio do seguinte vídeo, Tim Berners-Lee fala de como inventou a web. https://www.youtube.com/watch?v=OM6XIICm_qo Podemos atopalo subtítulado na seguinte ligazón: https://www.youtube.com/watch?v=UUpw-7_SWds

2. Aplicacións web

Na enxeñaría de software denomínase aplicación web a aquela ferramenta que os usuarios poden empregar accedendo a un servidor web a través de Internet ou dunha intranet mediante un navegador.

As aplicacións web son populares debido ao práctico que resulta o navegador web como cliente lixeiro, á independencia do sistema operativo, así como á facilidade para actualizar e manter aplicacións web sen distribuír e instalar software a miles de usuarios potenciais. Existen aplicacións como wikis, weblogs, tendas en liña, etc., que son exemplos bastante coñecidos de aplicacións web.

2.1 Vantaxes

- Aforro de tempo: pódense realizar tarefas sinxelas sen necesidade de descargar nin instalar ningún programa no noso ordenador.
- Non hai problemas de compatibilidade: xa que simplemente se precisa ter un navegador web actualizado para poder utilizalas.
- Non ocupan espazo no noso disco duro: xa que non se necesita gardar nin o propio programa nin os datos que se necesitan para operar con el.
- Actualizacións inmediatas: como o software o xestiona o propio desenvolvedor de software, cando nos conectamos á aplicación sempre estamos a usar a última versión que lanzou.
- Consumo de recursos baixo: dado que todo (ou gran parte) da aplicación non se atopa no noso ordenador, moitas das tarefas que realiza o software non consumen recursos nosos porque se realizan desde outra computadora, á cal estamos conectados vía Internet.
- Multiplataforma: pódense usar desde calquera sistema operativo porque soamente é necesario ter un navegador para usalas.
- Portables: é independente do ordenador onde se utilice (PC sobremesa, portátil) porque se accede a través dunha páxina web, polo que só é necesario un navegador e dispoñer de acceso a Internet.
- Disponibilidade alta: porque o servizo, aínda que hai ocasións no que se pode ver interrompido, ofrécese desde múltiples localizacións para asegurar a continuidade do mesmo.
- Os virus que poidan afectar ao noso ordenador non danan os datos porque estes están almacenados no servidor da aplicación.
- Colaboración: grazas a que o acceso ao servizo se realiza desde unha única ubicación, é sinxelo o acceso e a compartición de datos por parte de varios usuarios. Ten moito sentido, por exemplo, en aplicacións en liña de calendarios ou oficina.

2.2 Inconvenientes

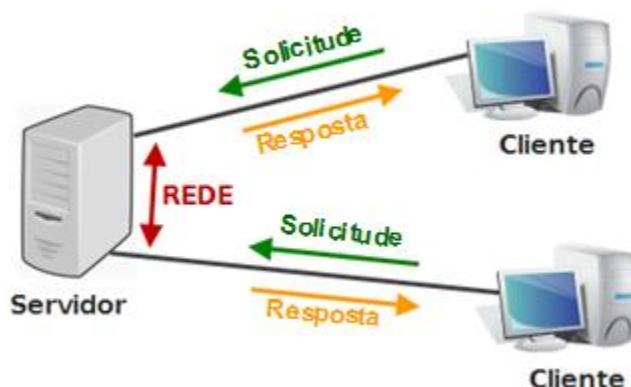
- Habitualmente ofrecen menos funcionalidade que as aplicacións de escritorio, debido a que as funcionalidades que se poden realizar desde un navegador poden ser máis limitadas que as que se poden realizar desde os sistema operativo.
- A dispoñibilidade depende dun terceiro, o provedor da conexión a Internet ou o que prové o enlace entre o servidor da aplicación e o cliente. Así que, a dispoñibilidade do servizo está supeditada ao provedor.

2.3 Modelos de arquitectura

O W3C (World Wide Web Consortium) define catro modelos de arquitectura para as aplicacións distribuídas: modelo cliente/servidor, modelo orientado a mensaxes, modelo orientado a servizos e modelo orientado a recursos. **O modelo cliente/servidor é a arquitectura máis estendida hoxe en día nas aplicacións web.**

Modelo cliente/servidor

Nesta arquitectura, por un lado está o cliente, que será o navegador web, e polo outro lado está o servidor, que é o servidor web. A lóxica da aplicación é compartida por ambos.



Modelo orientado a mensaxes

O Modelo Orientado a Mensaxes (MOM) céntrase en definir as mensaxes, a súa estrutura, a forma de transportalos, etc., sen importar o significado semántico de cada mensaxe ou a súa relación con outras mensaxes.

Modelo Orientado a servizos

O Modelo Orientado a Servizos (SOM) ocúpase dos aspectos da arquitectura relacionados co servizo e a acción.

O seu obxectivo principal é o de explicar as relacións entre un axente e os servizos que ofrece e solicita. O SOM baséase no MOM, pero centrado na acción, en lugar da mensaxe.

Modelo orientado a recursos

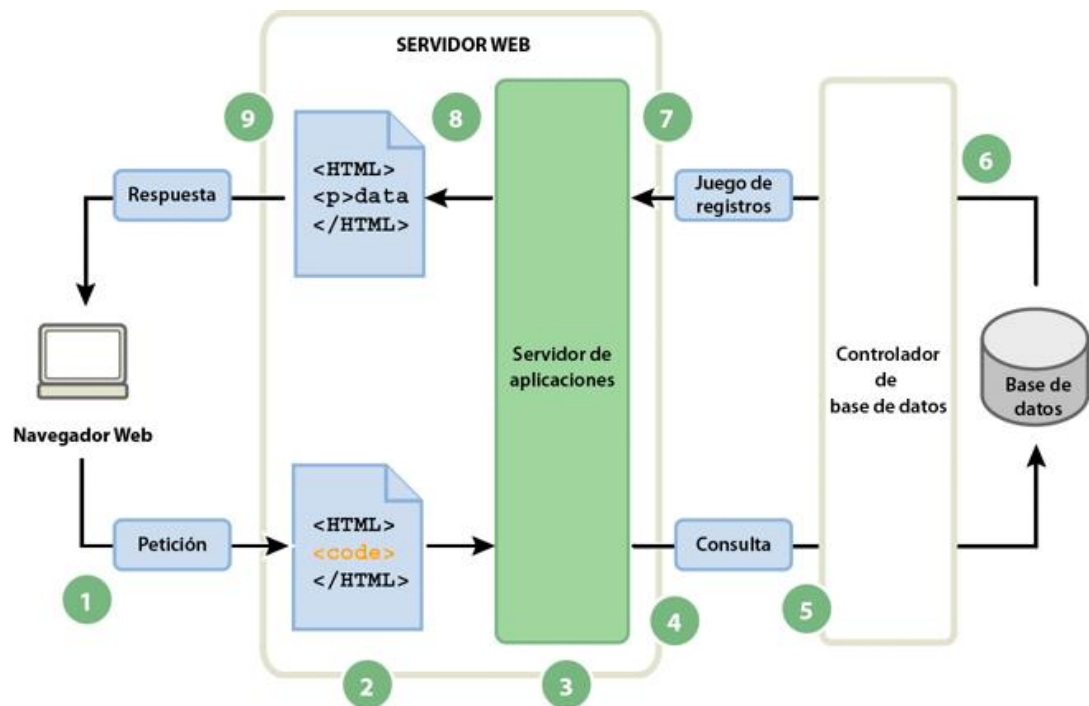
O Modelo Orientado a Recursos (ROM) céntrase nos aspectos da arquitectura relacionados cos recursos e as súas características principais (propietario, políticas asociadas, etc.), independentemente do papel que o recurso xogue no contexto dos servizos web.

2.4 Estrutura

Unha aplicación web que emprega o modelo cliente/servidor está estruturada habitualmente como unha aplicación en tres capas:

- Capa de presentación. É a que ve o usuario, presenta o sistema ao usuario, comunícalle a información e captura a información do usuario. Nunha aplicación web reside no cliente e as tecnoloxías empregadas habitualmente son HTML, CSS e Javascript, entre outras.
- Capa de negocio. É onde residen os programas que se executan, recíbense as peticións do usuario e envíanse as respostas tras o proceso. Nunha aplicación web reside no servidor e algunhas das tecnoloxías empregadas son PHP, Java Servlets o ASP, ASP.NET, CGI, ColdFusion, embPerl, Python ou Ruby on Rails.

- Capa de datos. É onde residen os datos e é a encargada de acceder aos mesmos. Nunha aplicación web reside no servidor, que pode ser fisicamente o mesmo que o que se ocupa da capa de negocio ou outro. Algunhas das tecnoloxías posibles para implementar esa capa son PostgreSQL, MySQL, Oracle DBMS, MongoDB ou Cassandra.



Na imaxe anterior pode verse claramente cal é o funcionamento dunha arquitectura web de 3 capas:

- 1. O usuario interactúa co navegador web que é o que realiza a solicitude de páxinas, que poden ser páxinas almacenadas (estáticas) ou creadas dinamicamente, con información aos servidores web.
- 2. O servidor web localiza a páxina e lla envía ao servidor de aplicacións para executar as instrucións necesarias.
- 3. O servidor de aplicacións busca as instrucións na páxina e execútaas.
- 4. O servidor de aplicacións precisará datos para poder executar as sentenzas, polo que envía a consulta ao controlador da base de datos.
- 5. O controlador executa a consulta na base de datos.
- 6. Unha vez resolta a consulta, envíase ao controlador o conxunto de rexistros resultantes da consulta.
- 7. O controlador pasa o conxunto de rexistros ao servidor de aplicacións.
- 8. O servidor de aplicacións insere os datos nunha páxina. Unha vez que esta é completa, pásallaa ao servidor web.
- 9. O servidor web envía a páxina finalizada ao navegador solicitante.

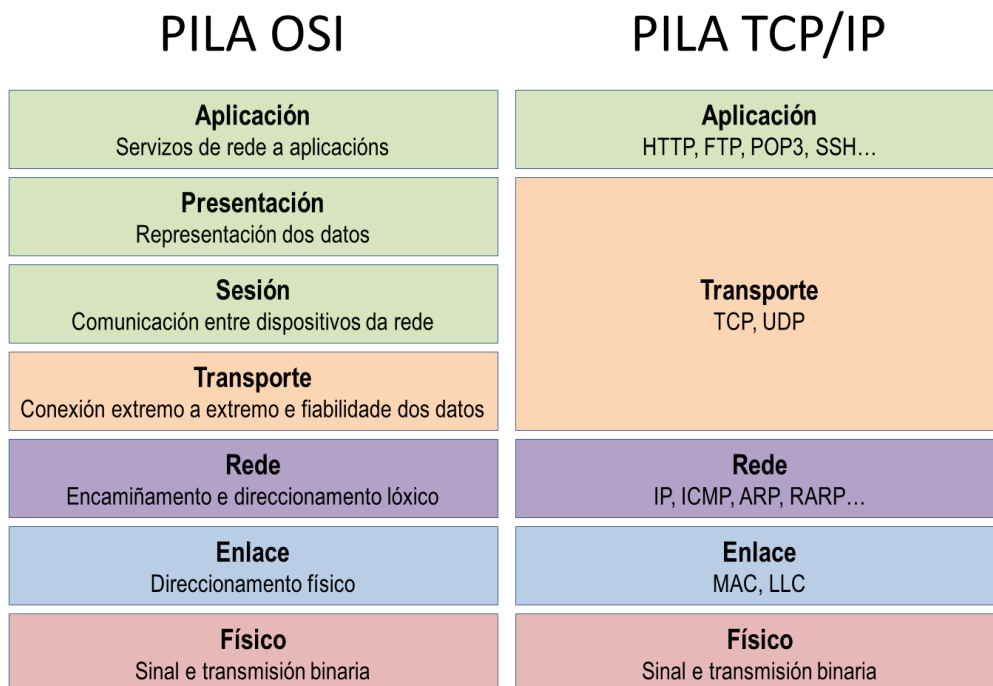
2.5 Software que intervéen

O tipo de software que rodea o uso e funcionamento dunha aplicación web no modelo cliente/servidor é o seguinte:

- **Navegador web:** é o software, aplicación ou programa que permite o acceso á web. A funcionalidade básica dun navegador é permitir a visualización de documentos, comunemente denominados páxinas web. Correspóndese coa capa de presentación dunha aplicación web. Co navegador interactúan os usuarios para poder facer uso dunha aplicación web.
- **Servidor web/servidor de aplicacións:** programas que procesan unha aplicación do lado do servidor. O servidor e o cliente comunícanse utilizando o protocolo HTTP para as comunicacións.
- **Linguaxes de programación / script:** os de script son un tipo de linguaxes de programación que non precisan ser compilados, senón que xeralmente son interpretados, de forma que as súas sentenzas se executan directamente, sen unha previa compilación. O conxunto de instrucións se denomina script, que é o que adoita acompañar a un documento HTML ou estar contido no seu interior. As instrucións do script execútanse antes de enviar o documento ao navegador, cando se carga o documento ou cando se produce algunha circunstancia, é dicir, cando ocorre un evento.
- **Xestor de base de datos:** é un conxunto de programas que permiten o almacenamento, modificación e extracción da información nunha base de datos, ademais de proporcionar ferramentas para engadir, borrar, modificar e analizar os datos.

3. Protocolo HTTP

Un servidor web opera mediante o protocolo HTTP, da capa de aplicación do Modelo OSI. HTTP é un protocolo simple, textual e sen estado para o intercambio de datos tipificados (obxectos) entre un cliente e un servidor web baseado en comandos e respostas.



Cada servidor ten un proceso que permanece á escoita nun porto TCP (por defecto, o 80), esperando conexións entrantes dos clientes. Cando se establece a conexión, os navegadores solicitan información ao servidor enviándolles un URL. O servidor responde cunha mensaxe que contén o estado da operación e o seu posible resultado. Todas as operacións poden achegar un obxecto ou recurso sobre o que actúan. Despois da transmisión libérase a conexión TCP (isto é que significa que http sexa un protocolo sen estado).

3.1 Mensaxes de petición

Os mensaxes de petición están formados por tres partes:

- Liña inicial de petición: método, URL, versión.
- Liña(s) de cabeceira.
- Corpo da mensaxe (opcional). Parámetros ou ficheiros a enviar ao servidor.

Os posibles métodos de petición son GET, POST, OPTIONS, HEAD, PUT, DELETE, TRACE, CONNECT e PATH.

Petición GET

Emprégase para obter calquera tipo de información do servidor. As peticións GET non teñen corpo da mensaxe. Permite enviar parámetros ao servidor na URI (ou URL) (coñecidos como Query String). Por exemplo:

```
http://sitiodeproba.com/novousuarioGET.php?nome=Emilio&apelido=Dapena
```

Nesta URL podemos distinguir varias partes:

- `http://sitiodeproba.com/novousuarioGET.php` é o enderezo web onde se atopa o script que procesará os datos.
- O símbolo `?` indica onde empezan os parámetros que se reciben desde o formulario que enviou os datos á páxina.
- As parellas `dato1=valor1`, `dato2=valor2`, etc. reflicten o nome e o valor dos campos enviados polo formulario.

O tamaño da información enviada estará limitada. Non se pode empregar para enviar arquivos ou realizar outras operacións que requiran enviar unha gran cantidade de datos ao servidor.

Petición POST

Emprégase para solicitar ao servidor que acepte información que se envía axunta nunha petición. As peticións POST envíanse no corpo da mensaxe. Os parámetros, polo tanto, non son visibles na URL.

3.2 Mensaxes de resposta

Os mensaxes de petición están formados por tres partes:

- Liña inicial de resposta (liña de estado): Versión HTTP, código de estado e texto explicativo.
- Liña(s) de cabeceira.

- Corpo da mensaxe (opcional). Determinado polo tipo de recurso solicitado.

Códigos de estado

Os códigos de estado son códigos que envían os servidores nas respostas HTTP. Informan ao cliente de cómo foi procesada a petición. Consisten nun código de 3 díxitos e se clasifican en función do primeiro, van acompañados dun texto explicativo. A continuación expoñemos os principais grupos de códigos de estado con algúns dos exemplos máis salientables.

- 100 - 199 (Informativo, Informational).
- 200 - 299 (Éxito, Successful):
 - 200 - OK (Resposta estándar para peticións correctas).
- 300 - 399 (Redirección, Redirection):
 - 304 - Not Modified (indica que a petición á URL non foi modificada desde que foi requerida por última vez).
- 400 - 499 (Erros do cliente, Client Error).
 - 404 - Not found (Non atopado).
 - 403 - Forbidden (A solicitude foi legal, pero o servidor rexeita respondela porque o cliente non ten os privilexios para facela).
 - 406 - Not Acceptable (O servidor non é capaz de devolver os datos en ningún dos formatos aceptados polo cliente).
- 500 - 599 (Erros no servidor, Server Error)
 - 503 - Service Unavailable (O servidor non pode responder á petición do navegador porque está conxestionado ou realizando tarefas de mantemento).



Para un listado de todos os códigos de estado das respostas HTTP pódese consultar o RFC-1945 <https://tools.ietf.org/html/rfc1945> no seu apartado 9: “Status Code Definitions”.

3.3 Tipos MIME

Inicialmente as páxinas web contiñan texto e ligazóns (hipertexto), polo que HTTP era un protocolo que unicamente enviaba texto. Agora, HTTP permite enviar todo tipo de documentos, para o que emprega os tipos MIME (tipos de medios de Internet). Os tipos MIME son entón unhas especificacións definidas pola IANA para dar formato a mensaxes non ASCII.

Exemplos de tipos MIME habituais son `text/html`, `text/css` ou `video/mpeg`.



Pódense consultar todos os tipos mime na especificación oficial do IANA <http://www.iana.org/assignments/media-types/media-types.xhtml>