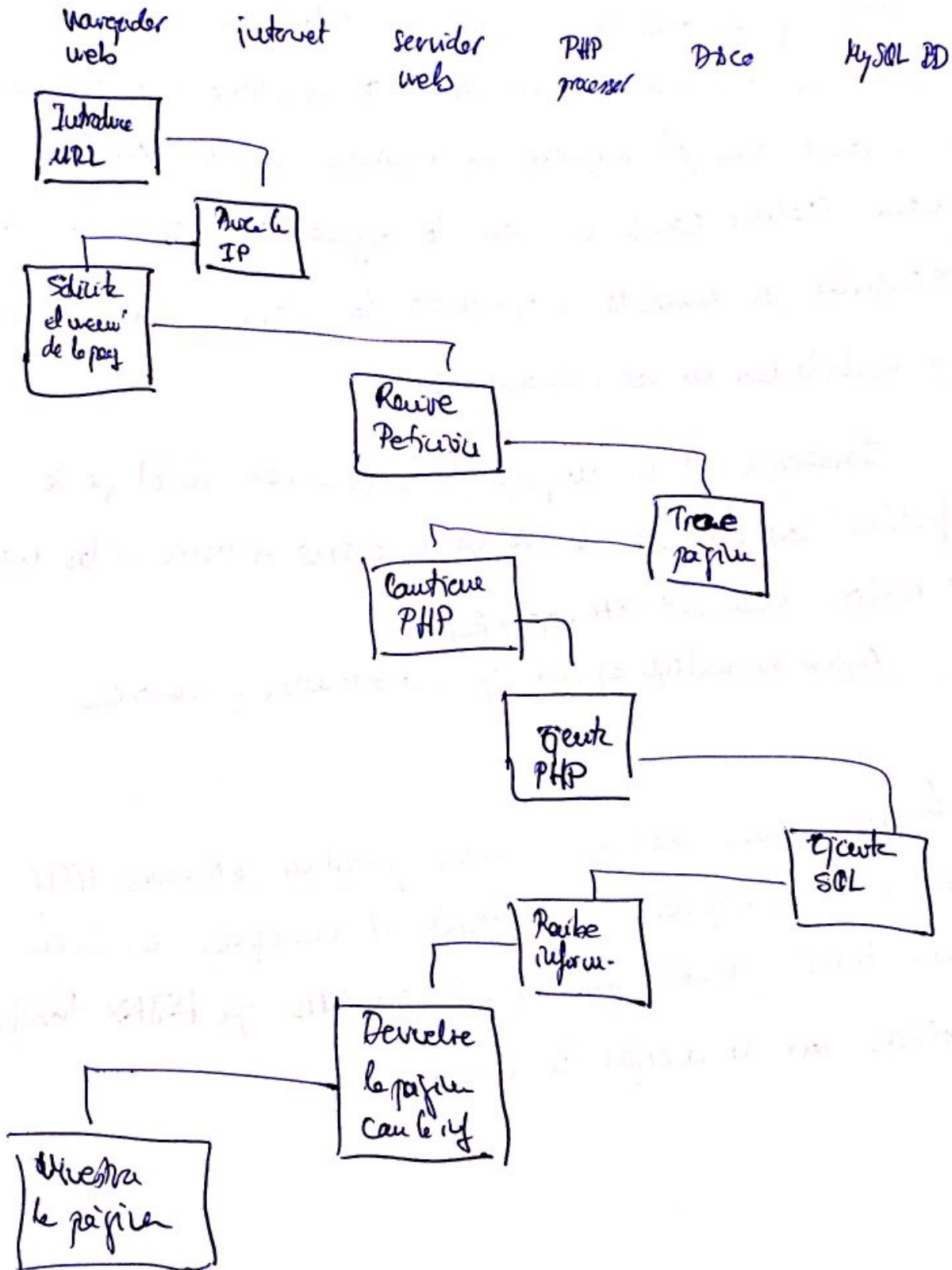


JUNIO 12-13

Ex 1 : Esquema de intranet ~~cliente~~ servidor web y navegador en caso de solicitud de contenidos dinámicos.



Ej 2 Principales diferencias entre aplicaciones web "clásicas" y aplicaciones AJAX.

AJAX (Asynchronous JavaScript And XML) es una técnica de desarrollo web que permite crear aplicaciones interactivas.

Estas aplicaciones se ejecutan en el lado del cliente, es decir, en el navegador mientras se mantiene una comunicación asincrónica con el servidor en segundos planos. Así se pueden realizar cambios en la página sin necesidad de recargarla por completo mejorando la interactividad, velocidad y usabilidad en las aplicaciones.

JavaScript es un lenguaje de programación en el que se efectúan las llamadas de Ajax mientras el acceso a los datos se realiza mediante XMLHttpRequest.

Ajax es un protocolo y una estándares abiertos

① Las aplicaciones clásicas mandan y reciben peticiones HTTP grandes a el navegador. Con AJAX el navegador envía una petición HTTP y recibe una respuesta en XML que AJAX transfiere en HTML para la interfaz de usuario.



② Por otro lado, en las aplicaciones clásicas el modelo de aprendizaje es  síncrono , en cambio en  ASAX  es  asíncrono .

③ Respecto al flujo de trabajo, en las aplicaciones clásicas, el usuario va de una página a otra con la recarga de páginas completas. En  ASAX , el flujo se define mediante  scripts  en el cliente y la comunicación con el servidor realizando  cambios parciales en la página . En consecuencia las peticiones no son registradas de forma automática en el historial.

④ Respecto al tráfico en la red,  ASAX  tiene  una carga leve inicial  y después es muy reducida. En las clásicas comienza con poca cantidad de datos inicial y va aumentando con el tiempo.

⑤ Respecto a los motores de búsqueda, con  ASAX  es  difícil indexar páginas con  <sup>este</sup>  contenido , al contrario de

## 8.3 Acceso a MySQL con PHP

### Esquema:

Repetir las veces necesarias

- 1° Conexión. ~~conectar~~
- 2° Seleccionar la BD
- 3° Construir la consulta
- 4° Ejecutar la consulta
- 5° Recuperar los datos y construir la pag. web
- 6° Desconectar

### Código

<?php

① Conexión

```
{ $db_conn = mysql_connect('localhost', 'usuario', 'clave');  
  if (!$db_conn)  
    die("No se puede conectar con MySQL");
```

② Seleccionar BD

```
mysql_select_db('mi-base-de-datos');
```

// 3- Construir consulta y ejecutar consulta

```
$consulta = 'select * from tabla-alumnos';
```

```
$result = mysql_query($consulta);
```

```
if (!$result)
```

```
  die("Error en la consulta");
```

// 5 - Recuperar Datos

```
$num-filas = mysql_num_rows ($result);  
for ($i=0, $i <= $num-filas; $i++)  
{  
    $fila = mysql_fetch_row ($result)  
    echo "Nombre " . $fila[0] . " /n";  
}
```

// Descartar

```
mysql_close ('mi-base-de-datos');
```

Q. 4 Describe como se crean y como se eliminan las cookies con PHP. Pon un ejemplo.

Mantienen informacion sobre los usuarios y se intercambian durante la transferencia de las cabeceras antes del envio de la pagina HTML.

Crear:

```
setcookie ('nombre', 'valor', time() + 60*60*24*7, '/', 'Ruta',
```

un dia completo

Eliminar:

```
setcookie ('nombre', 'valor', time() - 250000, '/', 'Ruta')
```



5.5 Explica en qué consiste Digital Object Model (DOM). Puntaje.

Es una interfaz de plataforma que proporciona un conjunto estándar de objetos para representar documentos HTML, XHTML y XML; en un modelo de árbol pueden combinarse dichos objetos y una interfaz para manipularlos.

A través de DOM los programas pueden acceder y modificar el contenido de los documentos HTML y XML.

Ejemplo:

```
<script>
```

```
url = document.links[0].linknamedocumenthref;
```

```
document.write('to URL es' + url);
```

```
</script>
```

SEPT. 12-13

## 5.1 REPETIDO.

7. → 5.2 Diferencias y ventajas entre desarrollo web en el lado del servidor vs cliente.  
Las aplicaciones que se ejecutan en el lado del cliente se ejecutan con poca carga ~~com~~ computacional. Permiten interactividad y ser más rápidas (efectos de animación, gráficos, validaciones).  
En cambio, supone un tiempo de carga inicial al cargarse la página.

Las aplicaciones en el lado del servidor permiten devolver contenido dinámico al usuario que ha realizado una petición.  
Realizan ejantes consultas a bases de datos. Inconvenientemente, se tiene que realizar peticiones constantemente al servidor.

## 5.3 Estructura mensajes HTTP.

Las peticiones y respuestas tienen la siguiente estructura: }

↳ ① Línea inicial      } Petición  
                                     } Respuesta

↳ ② Línea de cabecera. Metadatos.

↳ ③ Cuerpo del mensaje. Datos que se intercambian.

## Ex 4 Manipulo de arquivos PHP.

→ Podemos abrir arquivos em local: `fopen('arquivo.txt');`

→ ~~Manejo~~ <sup>Manejo</sup> arquivos com as seguintes funções

↳ leitura  $\Rightarrow$  `$fh = fopen('arquivo.txt', 'r');`

↳ escrever  $\Rightarrow$  `fwrite($fh, $texto);`

↳ fechar  $\Rightarrow$  `fclose($fh);`

## Ex 5 AJAX. (Repetido)



Tramo 13-14

Q1 Enumera los aspectos más importantes en el desarrollo aplicaciones web.

Interfaz web, es decir las funcionalidades para ofrecer una experiencia interactiva (como AJAX)

Consideraciones técnicas: en qué clientes o S.O. se va a ejecutar la aplicación.

Estructura: Estructurar las 3 capas de la aplicación:

Primera capa: el navegador web

Segunda capa: tecnología de web dinámica (PHP, JSP, etc)

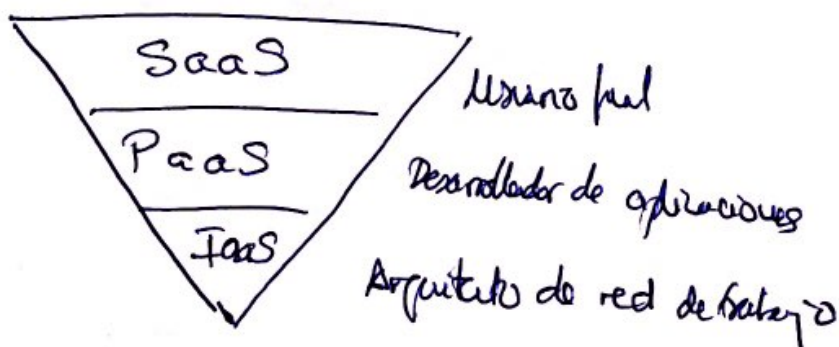
Tercera capa: base de datos.

Estado sin servidor: un modelo de computación que elimina la atención.

Q2. Hacerlo

Q3: Principales capas de servicios de plataforma Cloud Computing.

esta compuesto por 3 capas:



En la capa más alta se encuentra SaaS (Software as a Service), se garantiza por ofrecer una aplicación como un servicio, bajo demanda y se paga por el uso (no por tener licencias ni el software). Ejemplo: DropBox, Google Apps.

En la capa intermedia se encuentra la PaaS (Platform as a Service) contiene entorno software con herramientas de desarrollo para cada plataforma. Ejemplos comerciales como Google App Engine, Windows Azure.

En la capa final se encuentra la IaaS (Infrastructure as a Service), que permite el almacenamiento básico, capacidades de cómputo, servidores, sistemas de almacenamiento, etc. Ejemplo todo el hardware.

El desarrollador dispone de hardware virtual sobre el cual desarrolla las aplicaciones y servicios.

Ejemplo comercial es Amazon Web Services, Windows Azure.

#### Ej 4 Definición de servicios web

Es una tecnología que permite intercambiar datos entre aplicaciones gracias a un conjunto de protocolos y estándares.

De esta manera distintas aplicaciones de software escritas en lenguajes de programación distintos y sobre cualquier plataforma, pueden intercambiar datos entre redes de ordenadores como Internet.

#### Ej 5 Repetición pesos PySQL.



3.1 Enumera los aspectos más importantes en el desarrollo aplicaciones web.

- ① Diseño : debe contener una estructura, un contenido informativo, aspecto visual, consideraciones de compatibilidad y accesibilidad, funcionalidad, etc.
- ② Desarrollo en el ambiente del servidor : definir el lenguaje que se va a utilizar en el ambiente del servidor (PHP, Java, .Net).
- ③ Desarrollo en el ambiente del cliente : JavaScript, AJAX.
- ④ Bases de datos : MySQL, Oracle, PostgreSQL.

3.2 Explique la interacción entre las plataformas de Cloud Computing y las tecnologías web.

- Las tecnologías web sirven para acceder a los recursos de almacenamiento disponibles en Internet a través del navegador. El cloud Computing presta servicios a dichos recursos, de manera que los usuarios acceden a los recursos que proporcionan las tecnologías web y son ellos los que simplifiquen el uso de los recursos.

# Cloud Computing

Es un modelo de prestación de servicios de computo y tecnologías. El usuario accede a estos servicios bajo demanda y paga únicamente por el uso realzado. Se caracteriza por ser muy flexible y adaptable. El usuario tiene la sensación de tener recursos ilimitados.

## Características

- Aplicidad
- Costo
- Elasticidad y escalabilidad
- Independencia entre dispositivo y ubicación
- Rendimiento
- Mantenimiento

## Ventajas

- Reducción de costos (infraestructura, servidores de usuario, licencias soft., energía personal)
- Escalabilidad (uso simple, actualizaciones instantáneas, automat.)
- Prestaciones (disponibilidad permanente, capacidad ilimitada, adaptación elástica)
- Universalidad



8.2

- El Cloud Computing es un gran conjunto de computadores interconectados entre sí, están dispersos geográficamente lo que permite que las tecnologías web estén muy extendidas.
- El Cloud Computing tiene una escalabilidad elástica, esto permite que las tecnologías web puedan ser flexibles.

---

## Big Data

Es un concepto que hace referencia al almacenamiento de grandes cantidades de datos y a los procedimientos usados para encontrar patrones repetitivos dentro de esos datos.

Las dificultades más habituales son la recolección, almacenamiento, búsqueda, computación y visualización.

La tendencia de manipular tantos datos tiene el potencial de la creación de nuevos estadísticos y predictores

Herramientas para tratar  $\Rightarrow$  Hadoop, NoSQL, Amazon Integre, Cassandra, ...