

UNIVERSIDAD TECNOLÓGICA DE LEÓN

CAMPUS ACÁMBARO

Ingeniería en Tecnologías de la Información y Comunicación

ACTIVIDAD 1

Tecnologías web y arranque de proyecto

MATERIAS

Desarrollo de Aplicaciones Web

Administración de Proyectos II

PROFESORES

Rodolfo Martínez Puente

Mayra Neli Rivera Pizaña

ALUMNOS

|  |  |  |
| --- | --- | --- |
| 15000033 | Jonathan Jair Alfaro Sanchez | jonathan\_jair\_01@hotmail.com |
| 14001794 | Mary Carmen Crescencio Bernal | marylupe58@hotmail.com |
| 15000028 | Miguel Ángel Mandujano Barragán | miguel\_dark97@hotmail.com |
| 15002186 | Giovanni Misael Alfaro Sánchez | giovannimisael007@hotmail.com |

GRUPO

ITI-901

GENERACIÓN: 2017-2019

ACÁMBARO, GUANAJUATO. 19/Mayo/2018

**Tabla de contenido**

[INTRODUCCIÓN 1](#_Toc515193486)

[COMPETENCIA-OBJETIVO 2](#_Toc515193487)

[CONTENIDO 3](#_Toc515193488)

[TECNOLOGÍAS WEB 3](#_Toc515193489)

[Definición 3](#_Toc515193490)

[Lado servidor: Servidores 3](#_Toc515193491)

[Lado servidor: Programación 8](#_Toc515193492)

[Lado servidor: Gestores de Bases de Datos 1](#_Toc515193493)

[Lado cliente: Maquetado y Diseño 8](#_Toc515193494)

[Nuevas tecnologías web 16](#_Toc515193495)

[MEDIOS DIGITALES 34](#_Toc515193496)

[Audio 34](#_Toc515193497)

[Video 39](#_Toc515193498)

[Imagen 42](#_Toc515193499)

[PROTOCOLOS DE SEGURIDAD 49](#_Toc515193500)

[VULNERABILIDADES 52](#_Toc515193501)

[Tipos de vulnerabilidades 52](#_Toc515193502)

[ARRANQUE DEL PROYECTO 54](#_Toc515193503)

[Carta de inicio del proyecto 54](#_Toc515193504)

[Perfiles de los participantes y del administrador 55](#_Toc515193505)

[Matriz de responsabilidad 55](#_Toc515193506)

[Lista de factores de desempeño 57](#_Toc515193507)

[Esquema del ciclo de vida 58](#_Toc515193508)

[Justificación de las actividades del proceso 61](#_Toc515193509)

[Diagrama de Gantt 61](#_Toc515193510)

[Diagrama de ruta crítica 63](#_Toc515193511)

[REFLEXIONES 64](#_Toc515193512)

[PROBLEMAS PRESENTADOS 66](#_Toc515193513)

[BIBLIOGRAFÍA 67](#_Toc515193514)

[DIAGRAMA DE ACTIVIDADES 69](#_Toc515193515)

# INTRODUCCIÓN

Como sabemos, la tecnología evoluciona constantemente y no ha dejado de crecer en ningún momento. Como estudiantes del área de tecnologías podemos ver que, gracias a estos cambios la mejora de las aplicaciones se ha hecho más notable conforme pasa el tiempo, un claro ejemplo es el diseño, permitiéndonos hacer aplicaciones cada vez más atractivas al usuario.

La forma de trabajar en las empresas y para todo desarrollador en general se ha vuelto más sencilla, con la ayuda de frameworks, librerías y tecnologías que otros desarrolladores comparten con el mundo y que nos permiten obtener funcionalidades que ya no tenemos que hacer desde cero nosotros mismos.

Con el avance de la tecnología también nos vemos obligados a mejorar la manera de gestionar los proyectos, en donde la documentación juega un papel muy importante dentro de cualquier proyecto de software. Esta nos permite guardar una evidencia de lo que se quiere hacer y se está haciendo a lo largo de un proyecto.

Existen documentos importantes que se van haciendo antes de llegar a codificar el proyecto, como puede ser la matriz de comunicaciones, la planeación, matriz de responsabilidades, especificación de requerimientos, etc. todos ellos en conjunto nos ayudan a formar una buena base del proyecto.

# COMPETENCIA-OBJETIVO

El alumno de las materias de Administración de Proyectos II, Desarrollo de Aplicaciones Web y Optativa II durante la unidad 1 será competente para identificar las diferentes tecnologías web, así como los diferentes elementos que conforman el inicio de un proyecto, de tal manera que en la elaboración de una evaluación de procesos de producción extensa compuesta por un reporte de al menos 2000 palabras, obtenga como mínimo un 80% de desempeño de acuerdo a los criterios de la rúbrica de evaluación 1.

# CONTENIDO

## TECNOLOGÍAS WEB

### Definición

Las tecnologías web son una serie de herramientas que permiten visualizar a través de los navegadores web la diversidad de sitios que se encuentran en internet además de añadir la posibilidad de interactuar con estos, las distintas tecnologías web en conjunto (lenguajes, frameworks, librerías, etc.) permiten el desarrollo de diversos sitios web avanzados dependiendo de las tecnologías web usadas.

### Lado servidor: Servidores

* **APACHE**
* **Definición:**

Apache es un poderoso servidor web, cuyo nombre proviene de la frase inglesa “a patchy server” y es completamente libre, ya que es un software Open Source y con licencia GPL. Una de las ventajas más grandes de Apache, es que es un servidor web multiplataforma, es decir, puede trabajar con diferentes sistemas operativos y mantener su excelente rendimiento.

Desde el año 1996, es el servidor web más popular del mundo, debido a su estabilidad y seguridad. Apache sigue siendo desarrollado por la comunidad de usuarios desarrolladores que trabaja bajo la tutela de Apache Software Foundation.

* **Características:**
* Soporte de seguridad SSL y TLS.
* Puede realizar autentificación de datos utilizando SGDB.
* Puede dar soporte a diferentes lenguajes, como Perl, PHP, Python y tcl.
* **Usos:**

Apache es utilizado principalmente, para realizar servicio a páginas web, ya sean estáticas o dinámicas. Este estupendo servidor se integra a la perfección con otras aplicaciones, creando el famoso paquete XAMP con Perl, Python, MySQL y PHP, junto a cualquier sistema operativo, que por lo general es Linux, Windows o Mac OS.

* **Ventajas:**
* Es altamente configurable.
* Se desarrolla dentro de un proyecto HTT.
* Tiene amplia aceptación en la red.
* Posee licencia freeware gracias a su amplio nivel de capacitación, su costo y compatibilidad con sistemas operativos.
* Posee código abierto y es fácil de conseguir ayuda o soporte.
* **Desventajas:**
* Esta incluye formatos de configuración no estándar.
* No cuenta con una buena administración.
* Falta de integración.
* **IIS**
* **Definición:**

Los servicios de Internet Información Server (o IIS), son los servicios de software que admiten la creación, configuración y administración de sitios Web, además de otras funciones de Internet. Los servicios de Microsoft Internet Información Server incluyen el Protocolo de transferencia de noticias a través de la red (NNTP), el Protocolo de transferencia de archivos (FTP) y el Protocolo simple de transferencia de correo (SMTP).

Los servicios de Internet Información Server 5.0 (IIS) es el servicio Web de Windows 2000 que facilita la publicación de información en una intranet o en Internet; IIS permite autenticación robusta y segura de los usuarios, así como comunicaciones seguras vía SSL; además podemos crear contenido dinámico utilizando los componentes y secuencias de comandos del servidor para crear contenido dinámico independiente del explorador mediante páginas Active Server (ASP).

Además, en nuestro servidor IIS, instalaremos también un servidor PHP y la base de datos MSQL, de modo que podamos ejecutar páginas dinámicas PHP, así como enlazar tanto desde ASP como desde PHP con dicho gestor de base de datos; además también veremos cómo construir espacios seguros con SSL, así como espacios privados, etc.

* **Características:**
* IIS tiene la forma de asegurar los datos mediante SSL (SecureSocketsLayer). Esto proporciona un método para transferir datos entre el cliente y el servidor de forma segura, permitiendo también que el servidor pueda comprobar al cliente antes de que inicie sesión de usuario.
* La autenticación implícita que permite a los administradores autenticara los usuarios de forma segura a través de servidores de seguridad y proxy.
* IIS también es capaz de impedir que aquellos usuarios con direcciones IP conocidas obtengan acceso no autorizado al servidor, permitiendo especificar la información apropiada en una lista de restricciones.
* En cuanto a la seguridad, IIS tiene integrado el protocolo kerboros v5 (como le ocurre al sistema operativo).
* El almacenamiento de certificados se integra ahora con el almacenamiento CryptoAPI de Windows. Se puede utilizar el administrador de certificados de Windows para hacer una copia de seguridad, guarda y configura los certificados.
* La administración de la seguridad del servidor IIS es una tarea fácilmente ejecutable a base de asistentes para la seguridad.
* **Ventajas:**
* Es confiable, seguro y administrable en internet.
* Proporciona capacidades de servidor web integrado.
* Al momento de la instalación permite elegir sobre que servidor web va a correr.
* Desarrolla y es compatible con las aplicaciones beneficiándose con un único entorno de alojamiento de aplicaciones integrado con compatibilidad total.
* **Desventajas:**
* Tiende a limitarse.
* Posee vulnerabilidades.
* Este servidor no es multiplataforma, solo funciona bajo Windows.
* **TOMCAT**
* **Definición:**

Tomcat es un servidor Web con soporte para servlets y JSPs. No es un servidor de aplicaciones, como JBoss o JOnAS. Trae incluido el compilador Jasper, que compila JSPs convirtiéndolas en servlets. El motor de servlets de Tomcat a menudo se presenta en combinación con el servidor Web Apache. A partir de la versión 4.0, Tomcat utiliza el contenedor de servlets Catalina.

Tomcat puede funcionar como servidor Web por sí mismo. Al principio de su desarrollo existió la percepción de que la utilización de Tomcat de forma autónoma era sólo recomendable para entornos de desarrollo y entornos con mínimos requisitos de velocidad y gestión de transacciones. Actualmente ya no existe esa percepción y Tomcat es usado como servidor Web independiente en entornos con alto nivel de tráfico y alta disponibilidad.

El hecho de que Tomcat fue escrito en Java, hace posible que funcione en cualquier sistema operativo que disponga de la máquina virtual Java (también se puede utilizar con XAMPP). (Padrón, 2008)

* **Características:**
* Autenticación de acceso básico.
* Negociación de credenciales.
* HTTPS
* Alojamiento compartido.
* CGI o interfaz de entrada común.
* Servlets de Java.
* SSI
* Consola de administrador.
* **Ventajas:**
* Puede instalarse en muchos servidores web distintos (incluso IIS).
* Funciona en páginas JSP y Servlet.
* Es gratuito.
* Es de fácil configuración.
* **Desventajas:**
* Rendimiento bajo.
* No posee buena administración.
* **Tabla comparativa**

|  |  |  |
| --- | --- | --- |
| Nombre | Descripción | Características |
| Apache | Es un servidor web bastante potente y además flexible que puede funcionar en la mayoría de las plataformas y entornos, debido a que utiliza tantas plataformas ocasiona que a menudo sean necesarias ciertas características o funcionalidades. | Es un servidor completamente gratuito, de código abierto por lo cual se puede utilizar en cualquier plataforma Apache es una muestra, al igual que el Sistema Operativo Linux (un Unix desarrollado inicialmente para PC), de que el trabajo voluntario y cooperativo dentro de internet es capaz de producir aplicaciones de calidad profesional difíciles de igualar. |
| IIS | Es el Servidor Web de Microsoft, el IIS es el motor que ofrece dicha compañía de manera profesional, gracias a él es posible programar en ASP las cuales vienen a ser algo similares al PHP, este servidor tiene componentes programables desde ASP permitiendo acceder a cada uno de sus módulos para realizar una tarea específica. | Este tipo de servidor solo lo llevan sólo los sistemas Windows, Windows 2000 Professional, Windows 2000 Advanced Server, Windows XP Professional, los cuentan con las versiones 4.0, 5.0 y 5.1, Este sistema está en fase beta y se puede descargar desde la misma página de Microsoft, sólo es necesario registrarse. |
| TOMCAT | Apache Tomcat es un contenedor de servlets fue desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat incluye las especificaciones de los servlets y de JavaServer Pages (JSP) de Oracle Corporation | Tomcat es de software libre, cuenta con conexiones permanentes, soporte virtual, autentificación, envió de archivo, SSL, PHP, fue desarrollado en java. |

Se consideró que la mejor opción para utilizar es el servidor apache porque, es el más utilizado a nivel mundial, es un proyecto de código abierto, destaca demasiado por su seguridad y rendimiento y uso gratuito multiplataforma, por esa razón hay versiones para los sistemas operativos más importantes.

### Lado servidor: Programación

* **PHP**
* **Definición:**

PHP es un lenguaje de código abierto muy popular, adecuado para desarrollo web y que puede ser incrustado en HTML. Es popular porque un gran número de páginas y portales web están creadas con PHP.

* **Características:**
* **Gran extensión de documentación:** Gran cantidad de documentación desarrollada por la cantidad de usuarios que usan PHP.
* **Rapidez:** PHP no solo es rápido al ser ejecutado, sino que no genera retrasos en la máquina, por esto no requiere grandes recursos del sistema. PHP se integra muy bien junto a otras aplicaciones, especialmente bajo ambientes Unix.
* **Estabilidad:** PHP utiliza su propio sistema de administración de recursos y posee de un sofisticado método de manejo de variables, conformando un sistema robusto y estable.
* **Seguridad:** PHP maneja distintos niveles de seguridad, estos pueden ser configurados desde el archivo “**.ini”.**
* **Simplicidad:** Usuarios con experiencia en C y C++ podrán utilizar PHP rápidamente. Además, PHP dispone de una amplia gama de librerías, y permite la posibilidad de agregarle extensiones. Esto le permite su aplicación en múltiples áreas, tales como encriptado, gráficos, XML y otras.
* **Ventajas:**
* PHP corre en (casi) cualquier plataforma utilizando el mismo código fuente.
* La sintaxis de PHP es similar a la del C, por esto cualquiera con experiencia en lenguajes del estilo C podrá entender rápidamente PHP.
* PHP es completamente expandible y modificable. Está compuesto de un sistema principal, un conjunto de módulos y una variedad de extensiones de código.
* Muchas interfaces distintas para cada tipo de servidor. PHP actualmente se puede ejecutar bajo Apache, IIS, AOLServer, Roxen yTHTTPD. Otra alternativa es configurarlo como módulo CGI.
* Permite la interacción con gran cantidad de motores de bases de datos tales como MySQL, MS SQL, Oracle, Informix, PostgreSQL, etc.
* PHP es Open Source, (código abierto) esto significa que no depende de ninguna compañía comercial y que no requiere de licencias.
* **Desventajas:**
* Para poder ver y testear las páginas que vayamos creando es necesario disponer de un servidor web que soporte PHP.
* Al ser interpretado en el servidor, es más fácil que se colapse cuando el número de peticiones de descarga de páginas aumenta.
* Parte del contenido de las páginas puede no ser accesible a los navegadores, dificultando el posicionamiento de las páginas.
* **ASP**
* **Definición:**

El lenguaje ASP (Active Server Pages) se trata de un lenguaje de programación web desarrollado por Microsoft en 1996 para poder crear sitios web dinámicos. Como el PHP es un lenguaje interpretado en el lado del servidor.

La extensión de los archivos desarrollados en este lenguaje es .asp. El código ASP puede ser utilizado junto al código HTML. Se puede considerar como el estándar de desarrollo web dinámico para los servidores web ISS de Microsoft.

* **Características:**
* Se puede mezclar código ASP con código HTML y código de script en cliente, sin tener que compilarlos por separado. Esto facilita la construcción de páginas ASP, siendo posible escribirlas en un simple editor de textos (Notepad, Word, etc.).
* El código ASP se ejecuta en el servidor web, produciendo como salida HTML puro (junto con código de script en el cliente, si así se precisa).
* Permite usar componentes escritos en otros lenguajes de programación, como Visual Basic, Delphi, etc., que pueden ser llamados desde los propios guiones ASP.
* Permite acceder a las bases de datos de una forma rápida y sencilla, siendo este uno de sus principales usos.
* Tiene persistencia de variables en memoria entre distintas visualizaciones de página, permitiendo con ello manejar sesiones de usuario, con lo que se resuelve uno de los mayores problemas de la programación web, derivado de la no orientación a conexión del protocolo HTTP.
* **Ventajas:**
* Cuenta con códigos prediseñados lo cual da una mayor facilidad a la hora de diseñar una página web.
* El mejor sistema para programadores que conozcan Visual Basic.
* Fácil de aprender para los que no conozcan Visual Basic.
* Gratuito con el sistema operativo.
* Existen cientos de componentes para ASP.
* **Desventajas:**
* La principal desventaja es que solo funciona en plataforma Windows.
* El programador no lleva el control de las aplicaciones porque ya están prediseñadas.
* Es de licencia propietaria y el costo es muy elevado.
* **JSP**
* **Definición:**

Es un lenguaje para la creación de sitios web dinámicos, acrónimo de Java Server Pages. Está orientado a desarrollar páginas web en Java. JSP es un lenguaje multiplataforma. Creado para ejecutarse del lado del servidor. JSP fue desarrollado por Sun Microsystems. Comparte ventajas similares a las de ASP.NET desarrollado para la creación de aplicaciones web potentes.

* **Características:**
* Conjunta el poder de Java en el servidor y la flexibilidad de HTML en el browser.
* No sólo se puede utilizar HTML, sino también XML o WML.
* Hace más fácil reusar componentes con JavaBeans los cuales realizan tareas más específicas.
* Su función es saber cómo procesar una solicitud para crear una respuesta.
* Soporta contenido dinámico que refleja las condiciones del mundo real.
* Es más rápido y fácil crear aplicaciones de web.
* Capaz de instanciar cualquier clase de Java.
* **Ventajas:**
* Es un lenguaje totalmente escrito
* Fuerte capacidad de manipulación de gráficos
* Cargas de APIs: Interfaz de programación de aplicaciones
* Es open source.
* La ventaja fundamental en que tenemos toda la ventaja del lenguaje java a nuestro alcance: Reusabilidad, robustez y multiplataforma.
* **Desventajas:**
* No posee una interfaz con muchas posibilidades, ya que la interfaz web es limitada.
* La codificación de interfaces elaboradas lleva muchas líneas de código, y la mayoría debe hacerse usando scripts (javascript).
* La carga de la interfaz es más lenta que la de una aplicación de escritorio.
* La mayor parte de la lógica de la aplicación se ejecuta en el servidor, por lo que se corre el riesgo de sobrecargar de trabajo al mismo.
* La aplicación no está disponible si ocurre algún problema con el servidor o con la conexión de red.
* **Tabla comparativa**

|  |  |  |  |
| --- | --- | --- | --- |
| Lenguaje | Características | Fortalezas | Debilidades |
|  | * Posibilidad de integrar código PHP en HTML sin llamar a un archivo externo. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera el HTML resultante. * PHP es completamente expandible y modificable. * PHP no solo es rápido al ser ejecutado, sino que no genera retrasos en la máquina, por esto no requiere grandes recursos del sistema. * Usuarios con experiencia en C y C++ podrán utilizar PHP rápidamente. | * Su sintaxis es muy similar a otros lenguajes * Rápido * Multiplataforma * Maneja base de datos * Bastante documentado * Libre y gratuito. * No requiere definición de variables * Puede ser combinado junto a HTML * Tiene muchos frameworks que facilitan el desarrollo en este lenguaje. * Muchos servicios de alojamiento web tienen PHP. | * Para poder ver y testear las páginas que vayamos creando es necesario disponer de un servidor web que soporte PHP. * Al ser interpretado en el servidor, es más fácil que se colapse cuando el número de peticiones de descarga de páginas aumenta. * Parte del contenido de las páginas puede no ser accesible a los navegadores, dificultando el posicionamiento de las páginas. |
| ASP | * Se puede mezclar código ASP con código HTML y código de script en cliente, sin tener que compilarlos por separado. Esto facilita la construcción de páginas ASP, siendo posible escribirlas en un simple editor de textos (Notepad, Word, etc.). * El código ASP se ejecuta en el servidor web, produciendo como salida HTML puro (junto con código de script en el cliente, si así se precisa). * Permite usar componentes escritos en otros lenguajes de programación, como Visual Basic, Delphi, etc., que pueden ser llamados desde los propios guiones ASP. * Permite acceder a las bases de datos de una forma rápida y sencilla, siendo este uno de sus principales usos. | * Cuenta con códigos prediseñados lo cual da una mayor facilidad a la hora de diseñar una página web. * El mejor sistema para programadores que conozcan Visual Basic. * Fácil de aprender para los que no conozcan Visual Basic. * Gratuito con el sistema operativo. * Existen cientos de componentes para ASP. | * La principal desventaja es que solo funciona en plataforma Windows. * El programador no lleva el control de las aplicaciones porque ya están prediseñadas. * Es de licencia propietaria y el costo es muy elevado. |
|  | * Conjunta el poder de Java en el servidor y la flexibilidad de HTML en el browser. No sólo se puede utilizar HTML, sino también XML o WML. * Soporta contenido dinámico que refleja las condiciones del mundo real. * Es más rápido y fácil crear aplicaciones de web. * Capaz de instanciar cualquier clase de Java. | * Es un lenguaje totalmente escrito * Fuerte capacidad de manipulación de gráficos * Cargas de APIs: Interfaz de programación de aplicaciones * Es open source. * La ventaja fundamental en que tenemos toda la ventaja del lenguaje java a nuestro alcance: Reusabilidad, robustez y multiplataforma. | * No posee una interfaz con muchas posibilidades, ya que la interfaz web es limitada. * La codificación de interfaces elaboradas lleva muchas líneas de código. * La carga de la interfaz es más lenta que la de una aplicación de escritorio. * La mayor parte de la lógica de la aplicación se ejecuta en el servidor, por lo que se corre el riesgo de sobrecargar de trabajo al mismo. * La aplicación no está disponible si ocurre algún problema con el servidor o con la conexión de red. |

* **Elección del lenguaje de programación**

En nuestra opinión PHP es el mejor lenguaje para trabajar, ya que es rápido, libre y se actualiza constantemente con nuevas versiones y complementos, las desventajas que tiene se pueden mitigar con la ayuda de múltiples herramientas, hay mucha documentación sobre su uso ya que tiene una comunidad muy amplia, dando la posibilidad de compartir dudas, o errores con nuestro código, puede ser usado casi con cualquier máquina, ya que no necesita muchas especificaciones. Además, puede ser usado en cualquier sistema operativo, esto le da una ventaja muy grande contra los demás lenguajes.

### Lado servidor: Gestores de Bases de Datos

Un Sistema Gestor de Bases de Datos (SGBD) o DGBA (Data Base Management System) es un conjunto de programas no visibles que administran y gestionan la información que contiene una base de datos. A través de él se maneja todo acceso a la base de datos con el objetivo de servir de interfaz entre ésta, el usuario y las aplicaciones.

Gracias a este sistema de software invisible para el usuario final, compuesto por un lenguaje de definición de datos, un lenguaje de manipulación y de consulta, es posible gestionar los datos a distintos niveles. Tanto almacenar, modificar y acceder a la información como realizar consultas y hacer análisis para generar informes. (PowerData Especialistas en Gestión de Datos, 2016)

* **MySQL**



Figura 1 Logotipo de MySQL

* **Definición:**

MySQL es un gestor de base de datos completamente desarrollado en lenguaje C/C++, lo que ofrece una estabilidad de trabajo impresionante. También, cuenta entre sus características, con una excelente capacidad de integración con diferentes entornos de desarrollo de software y de aplicaciones cliente/servidor, por lo tanto, es muy popular entre los programadores de aplicaciones web y entre administradores de base de datos en todo el mundo. Es un sistema de gestión de base de datos relacional, multiusuario con más de seis millones de instalaciones. MySQL AB desarrolla MySQL como software libre en un esquema de licenciamiento dual.

* **Características:**
* Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
* Soporta gran cantidad de tipos de datos para las columnas.
* Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc).
* Gran portabilidad entre sistemas.
* Soporta hasta 32 índices por tabla.
* Gestión de usuarios y contraseñas, manteniendo un muy buen nivel de seguridad en los datos.
* Condición de open source de MySQL hace que la utilización sea gratuita y se puede modificar con total libertad.
* Se puede descargar su código fuente. Esto ha favorecido muy positivamente en su desarrollo y continuas actualizaciones.
* Es una de las herramientas más utilizadas por los programadores orientados a Internet.
* Infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación.
* MYSQL, es el manejador de base de datos considerado como el más rápido de Internet.
* Gran rapidez y facilidad de uso.
* Fácil instalación y configuración.
* **Ventajas:**
* MySQL es OpenSource; “Al igual que PHP, su principal ventaja reside en que es una base de datos gratuita” (Puertas, 2004).
* Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
* Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una máquina con escasos recursos sin ningún problema.
* Facilidad de configuración e instalación.
* Soporta gran variedad de Sistemas Operativos.
* Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
* El software MySQL usa la licencia GPL.
* **Desventajas:**
* Un gran porcentaje de las utilidades de MySQL no están documentadas.
* No es intuitivo, como otros programas (ACCESS).
* Otro problema común es, el tamaño de la base de datos. Aunque en teoría es escalable hasta 8 TB, MySQL no puede trabajar de manera eficiente con grandes bases de datos.
* **MS SQL Server**

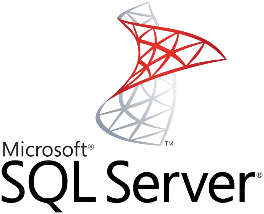


Figura 2 Logotipo de Microsoft SQL Server

* **Definición:**

SQL Server es un sistema gestor de base de datos relacionales producido por Microsoft. Es un sistema cliente/servidor que funciona como una extensión natural del sistema operativo Windows. Entre otras características proporciona integridad de datos, optimización de consultas, control de concurrencia y backup y recuperación.

Es relativamente fácil de administrar a través de la utilización de un entorno gráfico para casi todas las tareas de sistema y administración de bases de datos. Utiliza servicios del sistema operativo Windows para ofrecer nuevas capacidades o ampliar la base de datos, tales como enviar y recibir mensajes y gestionar la seguridad de la conexión. Es fácil de usar y proporciona funciones de almacenamiento de datos que sólo estaban disponibles en Oracle y otros sistemas gestores de bases de datos más caros.

Es un sistema de gestión de bases de datos relacionales basado en el lenguaje Transact-SQL, capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea.

* **Características:**
* Soporte de transacciones.
* Escalabilidad, estabilidad y seguridad.
* Soporta procedimientos almacenados.
* Incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
* Permite trabajar en modo cliente-servidor, donde la información y datos se alojan en el servidor y los terminales o clientes de la red sólo acceden a la información.
* Además, permite administrar información de otros servidores de datos.
* **Ventajas:**
* SQL Server nos permite olvidarnos un poco de los ficheros que forman la base de datos. En Access tenemos que abrir el fichero .mdb para ver el contenido. Aquí abrimos la consola de SQL Server y tenemos todas las bases de datos de todos nuestros programas.
* Si trabajamos en una red local nos permite agregar otros servidores de SQL Server.
* Ofrece una potente forma de unir SQL e Internet.
* **Seguridad:** SQL permite administrar permisos a TODO. Permisos a nivel de servidor, seguridad en tablas, permitir o no lectura, escritura, ejecución, todo se puede configurar. O sea, que podemos permitir que alguien conecte su SQL al nuestro, pero sin embargo podemos decirle que no puede ver esta base de datos, pero otra sí.
* **Desventajas:**
* Utiliza mucho la memoria RAM para las instalaciones y utilización de software.
* No se puede utilizar como practicas porque se prohíben muchas cosas, tiene restricciones en lo particular.
* La relación, calidad y el precio está muy debajo comparado con Oracle.
* Tiene muchos bloqueos a nivel de página, un tamaño de página fijo y demasiado pequeño, una pésima implementación de los tipos de datos variables.
* **Oracle**



Figura 3 Logotipo de Oracle

* **Definición:**

Fabricado por Oracle Corporation, utiliza la arquitectura cliente/servidor. Ha incorporado en su sistema el modelo objeto-relacional, pero al mismo tiempo garantiza la compatibilidad con el tradicional modelo relacional de datos. Así ofrece un servidor de bases de datos híbrido. Es uno de los más conocidos y ha alcanzado un buen nivel de madurez y de profesionalidad. Se destaca por su soporte de transacciones, estabilidad y escalabilidad.

* **Características:**
* **Modelo relacional:** los usuarios visualizan los datos en tablas con el formato filas/columnas.
* Herramienta de administración gráfica intuitiva y cómoda de utilizar.
* **Control de acceso:** tecnologías avanzadas para vigilar la entrada a los datos.
* **Protección de datos:** seguridad completa en el entorno de producción y de pruebas y gestión de copias de seguridad.
* Lenguaje de diseño de bases de datos muy completo (PL/SQL): permite implementar diseños "activos", que se pueden adaptar a las necesidades cambiantes de negocio.
* **Alta disponibilidad:** escalabilidad, protección y alto rendimiento para la actividad empresarial.
* **Gestión de usuarios:** agilidad en los trámites, reducción de costes y seguridad en el control de las personas que acceden a las aplicaciones y a los sistemas.
* **Ventajas:**
* Oracle es el motor de base de datos objeto-relacional más usado a nivel mundial.
* Puede ejecutarse en todas las plataformas, desde una PC hasta un supercomputador.
* Oracle soporta todas las funciones que se esperan de un servidor "serio": un lenguaje de diseño de bases de datos muy completo (PL/SQL) que permite implementar diseños "activos", con triggers y procedimientos almacenados, con una integridad referencial declarativa bastante potente.
* Permite el uso de particiones para la mejora de la eficiencia, de replicación e incluso ciertas versiones admiten la administración de bases de datos distribuidas.
* Existe incluso una versión personal para Windows 9x, lo cual es un punto a favor para los desarrolladores que se llevan trabajo a casa.
* Oracle es la base de datos con más orientación hacía INTERNET.
* **Desventajas:**
* Las versiones más recientes de Oracle son la 11g, 10g, 9g, 8g, desde el lanzamiento original de la 8 se sucedieron varias versiones con correcciones, hasta alcanzar la estabilidad en la 8.0.3. El motivo de tantos fallos fue, al parecer, la remodelación del sistema de almacenamiento por causa de la introducción de extensiones orientadas a objetos.
* El mayor inconveniente de Oracle es quizás su precio. Incluso las licencias de Personal Oracle son excesivamente caras.
* Otro problema es la necesidad de ajustes. Un error frecuente consiste en pensar que basta instalar el Oracle en un servidor y enchufar directamente las aplicaciones clientes. Un Oracle mal configurado puede ser desesperantemente lento.
* También es elevado el costo de la información, y sólo últimamente han comenzado a aparecer buenos libros sobre asuntos técnicos distintos de la simple instalación y administración.
* **Comparativa:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | MySQL | MS SQL Server | Oracle |
| ACID | Depende | SI | SI |
| Integridad referencial | Depende | SI | SI |
| Transacciones | Depende | SI | SI |
| Unicode | SI | SI | SI |
| Interfaz | SQL | GUI, SQL, Various | GUI, SQL |
| Lenguaje Soportado | Many, including C, C#, C++, D, Java, Ruby y Objective C | Java, Ruby, Python, VB, .Net y PHP | Many, including C, C#, C++, Java, Ruby y Objective C |
| Sistema Operativo | Windows, Linux, OS X, FreeBSD, Solaris | Windows | Windows, Linux, Solaris, HP-UX, OS X, z/OS, AIX |
| Licencia | Código Libre | Propietario | Propietario |

Para tomar una decisión sobre cuál gestor es el mejor, considero hay muchos factores. Podría decir que Oracle es el mejor, pues tiene una gran empresa por detrás de él, por algo es de paga, ya que no cualquier gestor provee lo que él puede. Sin embargo, para el nivel en el que nos encontramos actualmente considero que MySQL es el mejor gestor, pues es libre, lo que lo hace más accesible al público y no se queda tan atrás en cuento a capacidades y funciones que provee.

### Lado cliente: Maquetado y Diseño

* **HTML5**
* **Definición:**

(*HyperText Markup Language*, versión 5) es la quinta revisión del lenguaje HTML. Esta nueva versión (aún en desarrollo), y en conjunto con CSS3, define los nuevos estándares de desarrollo web, rediseñando el código para resolver problemas y actualizándolo así a nuevas necesidades. No se limita solo a crear nuevas etiquetas o atributos, sino que incorpora muchas características nuevas y proporciona una plataforma de desarrollo de complejas aplicaciones web (mediante los APIs).

HTML5 está destinado a sustituir no sólo HTML 4, sino también XHTML 1 y DOM Nivel 2. Esta versión nos permite una mayor interacción entre nuestras páginas web y el contenido media (video, audio, entre otros) así como una mayor facilidad a la hora de codificar nuestro diseño básico.

* **Características:**
* Nuevas etiquetas semánticas para estructurar los documentos HTML, destinadas a remplazar la necesidad de tener una etiqueta <div> que identifique cada bloque de la página.
* Los nuevos elementos multimedia como <audio> y <video>.
* La integración de gráficos vectoriales escalables (SVG) en sustitución de los genéricos <object>, y un nuevo elemento <canvas> que nos permite *dibujar* en él.
* El cambio, redefinición o estandarización de algunos elementos, como <a>, <cite> o <menu>.
* MathML para fórmulas matemáticas.
* Almacenamiento local en el lado del cliente.
* Y otros muchos nuevos APIs que veremos a lo largo de los siguientes capítulos.
* **Ventajas:**
* **Nueva estructura de etiquetas mejorada**, esta nueva estructura permite definir por separado el encabezado, la barra de navegación, las secciones de la página web, los textos del sitio, los diálogos y el pié de página de los sitios web. Esta nueva estructura, incluso permite desarrollar blogs con gran facilidad.
* **Inclusión de las etiquetas video y audio**, dicha etiqueta soporta de manera eficiente y estable cualquier opción de ejecución de video y audio, sin generar errores o incluir código flash en nuestro sitio web.
* **Capacidad de realizar ejecuciones offline de las páginas web creadas con código HTML5**, lo que permite realizar aplicaciones de escritorio con este código tan versátil.
* **Incluye una nueva etiqueta de dibujo sobre la página web**, llamada canvas, que vuelve el proceso de crear dibujos en el sitio web tan fácil como dibujar con aplicaciones como Paint.
* **Eliminación total de las etiquetas obsoletas**, que tienen los estándares anteriores de HTML.
* **Ofrece la posibilidad de obtener un código más limpio** y fácil de depurar, que los códigos de los estándares anteriores.
* Es nativo, y por tanto independiente de plugins de terceros. Es decir, no pertenece a nadie, es opensource.
* Es más semántico, con etiquetas que permiten clasificar y ordenar en distintos niveles y estructuras el contenido. Además, incorpora metadatos de manera más formal, favoreciendo el posicionamiento SEO y la accesibilidad.
* El código es más simple lo que permite hacer páginas más ligeras que se cargan más rápidamente favoreciendo la usabilidad y la indexación en buscadores.
* Ofrece una compatibilidad mayor con los navegadores de dispositivos móviles.
* Incluye la etiqueta de dibujo canvas, que ofrece más efectos visuales.
* Ofrece soporte a codecs específicos.
* Posibilita la inserción de vídeos y audio de forma directa.
* Permite la geolocalización del usuario. Algo muy útil para el marketing móvil.
* Tiene la capacidad de ejecutar páginas sin estar conectado.
* Incorpora nuevas capacidades Javascript que aumentan la capacidad de almacenamiento. Frente a las cookies que dejaban almacenar algunos kilobytes, ahora se puede conseguir el almacenamiento de entre 5 y 10 megas, dependiendo de la plataforma. Además, se permiten múltiples Javascripts corriendo en paralelo en una misma página.
* Dispone de nuevas capacidades CSS3 como posibilidad de usar cualquier fuente o tipografía en HTML, columnas de texto, opacidad, transparencia, canales alpha, contraste, saturación, brillo, animaciones de transición y transformación, bordes redondeados, gradientes, sombras, etc.
* Permite realizar diseños adaptables a distintos dispositivos (web, tablets, móviles…)
* **Desventajas:**
* Formato de vídeo estándar:  por el momento no hay un formato de vídeo compatible con todos los navegadores web y cuando a una plataforma se suben 24 horas de vídeo cada minuto es importante minimizar el número de formatos a soportar.
* Control de streaming: el visor de vídeo en Flash permite acceder a partes en concreto del mismo a través de la manipulación de variables que Flash toma e interpreta via Actionscript y que HTML5 no tiene. La etiqueta vídeo de HTML5 muestra un archivo como quien enlaza una foto sin posibilidad de manejar el concepto streaming.
* HTML5 aún no cuenta con todas las funcionalidades necesarias para poder incrustar un vídeo en la red con todos los elementos que tiene YouTube actualmente: anotaciones, subtítulos, anuncios, relacionados… Además, tampoco permite reproducir vídeos HTML5 a pantalla completa.
* Acceso a cámara y micrófono: La etiqueta video de HTML5 sirve para ver videos, no para interactuar, mientras que Flash lleva una importante ventaja tecnológica al permitir grabar y hacer videoconferencias desde el browser.
* **CSS3**
* **Definición:**

CSS significa Cascade Style Sheets, también llamado Hojas de Estilo en Cascada. CSS es un lenguaje de marcado que se emplea para dar formato a un sitio web. Es decir, funciona en conjunto con los archivos HTML.

Es la última evolución del lenguaje de las Hojas de Estilo en Cascada y pretende ampliar la versión CSS2.1. Trae consigo muchas novedades altamente esperadas, como las esquinas redondeadas, sombras, gradientes, transiciones o animaciones, y nuevos layouts como multi-columnas, cajas flexibles o maquetas de diseño en cuadrícula (grid layouts).

* **Características:**
* **Atributo gradiente de colores en borde con CSS y Firefox:** Posibilidad de definir el un gradiente de color en el borde de los elementos con CSS, en un atributo no estándar de Firefox.
* **Bordes redondeados en CSS 3:** Las características de CSS 3 incluyen bordes redondeados, a través del atributo border-radius, que define la curvatura que debe tener el borde del elemento.
* **Múltiples imágenes de fondo con CSS:** Cómo conseguir que un elemento de la página tenga varias imágenes de fondo a la vez, con CSS básico y con características de CSS 3.
* **Colores RGBA en CSS 3:** Veremos qué son los colores RGBA y su notación, que se incluyen en la especificación de Hojas de Estilo en Cascada CSS 3.
* **Word-wrap y overflow-wrap en CSS 3:** Una propiedad de CSS 3 que sirve para romper las palabras que son demasiado largas y no caben enteras por la anchura de una caja. Atributos word-wrap y overflow-wrap.
* **Textos multi-columna con CSS 3:** CSS 3 incorpora nuevos atributos para que el navegador se encargue de producir texto multicolumna, es decir, que maquete directamente el texto en varias columnas sin tener que hacer nosotros nada.
* **Bordes con imágenes en CSS 3:** El atributo border-image y varios otros de CSS 3 harán posible la utilización de imágenes como bordes de los elementos de la página, sin código HTML especial, simplemente con hojas de estilo.
* **Sombras en CSS 3 con box-shadow:** Crear sombras en CSS3 con el atributo box-shadow. Por fin podremos aplicar sombras a los elementos de la página, sin usar imágenes, Javascript ni nada extra, simplemente con un atributo de CSS 3.
* **Resplandor exterior con CSS3:** Cómo realizar un elemento que tenga un resplandor exterior con CSS3 y la propiedad box-shadow.
* **Propiedad background-origin de CSS 3:** La propiedad de CSS 3 background-origin permite decidir la posición de la imagen de fondo con respecto al borde, padding o el contenido del elemento.
* **Atributos CSS3 overflow-x y overflow-y:** Descripción de los atributos de CSS3 overflow-x y overflow-y, que sirven para definir cómo renderizar un contenido cuando sobrepasa los límites de un contenedor en la horizontal o vertical.
* **Sombras en el texto con text-shadow de CSS:** Cómo aplicar sombras y otros efectos en los textos con CSS y el atributo text-shadow.
* **Ventajas:**
* Incluye nuevas propiedades especialmente en cuanto al aspecto gráfico (recordemos que CSS también es utilizado para describir estilos para medios de audio) aunque se prevean mejores sustanciales en otros medios como el de sonido.
* Permite lograr estilos y efectos visuales que antes sólo eran posibles por medio de tecnologías adicionales.
* Las novedades de CSS3 nos permiten ahorrarnos tiempo y trabajo al poder seguir varias técnicas (bordes redondeados, sombra en el texto, sombra en las cajas, etc) sin necesidad de usar un editor gráfico.
* También se habla del hecho de que con estas nuevas propiedades la carga de la página deberá descender pues el hecho de que muchos de los efectos estén ahora bajo control del navegador hará que los recursos visuales e imágenes que ahora empleamos ya no tengan razón de seguir siendo utilizados.
* **Desventajas:**
* El uso de las tablas nos permitía crear diseños complejos de forma mucho más sencilla que utilizando CSS, aunque CSS3 está intentando facilitar dicho trabajo.
* **JavaScript**
* **Definición:**

Es un lenguaje de programación que te permite crear contenido nuevo y dinámico, controlar archivos de multimedia, crear imágenes animadas y muchas otras cosas más. (Aunque, no todo, pero es increíble lo que puedes llegar a hacer con tan solo unas pocas líneas de código de JavaScript).

* **Características:**
* Es simple, no hace falta tener conocimientos avanzados de programación para poder hacer un programa en JavaScript.
* Maneja objetos dentro de nuestra página Web y sobre ese objeto podemos definir diferentes eventos. Dichos objetos facilitan la programación de páginas interactivas, a la vez que se evita la posibilidad de ejecutar comandos que puedan ser peligrosos para la máquina del usuario, tales como formateo de unidades, modificar archivos etc.
* Es dinámico, responde a eventos en tiempo real. Eventos como presionar un botón, pasar el puntero del mouse sobre un determinado texto o el simple hecho de cargar la página o caducar un tiempo. Con esto podemos cambiar totalmente el aspecto de nuestra página al gusto del usuario, evitándonos tener en el servidor una página para cada gusto, hacer cálculos en base a variables cuyo valor es determinado por el usuario, etc.
* Su sintaxis es similar a la usada en Java y C, al ser un lenguaje del lado del cliente este es interpretado por el Navegador, no se necesita tener instalado ningún Framework.
* **Ventajas:**
* Es un lenguaje muy sencillo.
* Es rápido, por lo tanto, tiende a ejecutar las funciones inmediatamente.
* Cuenta con múltiples opciones de efectos visuales.
* Es soportado por los navegadores más populares y es compatible con los más modernos, incluyendo iPhone, móviles y PS3.
* Es muy versátil, puesto que es muy útil para desarrollar páginas dinámicas y aplicaciones web.
* Es una buena solución para poner en práctica la validación de datos en un formulario.
* Es multiplataforma, puede ser ejecutado de manera híbrida en cualquier sistema operativo móvil.
* Es el único lenguaje que permite trabajar modo FullStack en cualquier tipo de desarrollo de programación.
* **Desventajas:**
* En el FrontEnd sus códigos son visibles, por lo tanto pueden ser leídos por cualquier usuario.
* Tiende a introducir gran cantidad de fragmentos de código en los sitios web.
* Sus opciones 3D son limitadas, si se quiere utilizar este lenguaje de programación para crear un juego, deben emplearse otras herramientas.
* No es compatible en todos los navegadores de manera uniforme.
* Los usuarios tienen la opción de desactivar JavaScript desde su navegador.
* Sus scripts son limitados por razones de seguridad y no es posible realizar todo con JavaScript, por lo tanto, es necesario complementarlo con otros lenguajes evolucionados y más seguros. Esta es una de las características de JavaScript que algunos expertos lo contemplan como una ventaja y otros como una desventaja.
* **Tabla comparativa:**

|  |  |  |
| --- | --- | --- |
| Nombre y Características | Explicación | ¿Cuál es mejor? |
| HTML5  Lenguaje para construir las páginas web. | Nos aporta la estructura básica de los sitios. Es el lenguaje famoso por sus etiquetas conocido por todos los que se han inmerso en el desarrollo de páginas web. | No se puede comparar estas herramientas, ya que son partes complementarias para trabajar sobre un mismo proyecto web y cada uno se usa con funciones diferentes lo que hace imposible poder catalogar a una como la más importante. |
| CSS3  Lenguaje para estilizar las páginas web. | Se utiliza para controlar la presentación, el formato, y el diseño de nuestra aplicación. Gracias a ella podemos dar forma al estilo que queremos que se muestre. |
| JavaScript  Lenguaje para *****programar*****las páginas web. | Se utiliza para controlar el comportamiento de los diferentes elementos, esto es, aporta funcionalidad, dinámica, a nuestros elementos. |

### Nuevas tecnologías web

* **POLYMER**
* **Definición:**

Polymer es una biblioteca liviana que lo ayuda a aprovechar al máximo los componentes web.

Con Web Components, puede crear elementos personalizados reutilizables que interactúen sin problemas con los elementos integrados del navegador, o divida su aplicación en componentes de tamaño adecuado, haciendo que su código sea más limpio y menos costoso de mantener.

* **Tecnologías que emplea:**
* Custom Elements (Elementos personalizados)
* HTML Templates (Plantillas HTML)
* Shadow DOM (DOM oculto)
* HTML Imports
* **Ventajas de uso:**
* **Polymer te permite usar Web Components:** Polymer es un conjunto de librerías especiales, mejor conocidas como Polyfills. Estos tienen el poder de habilitar funcionalidades adicionales a los navegadores. Los Polyfills de Polymer nos permiten usar estas tecnologías del futuro en los principales navegadores de hoy.
* **Puedes crear tus propios Custom Elements:** Esto te permite guardar las partes de tu aplicación en etiquetas HTML5. Y así puedes desarrollar tu aplicación en bloques que puedes rehusar una y otra vez. Un Custom Element funciona de manera independiente del resto de tu aplicación y contiene todo lo relacionado a un elemento de la interfaz, un ejemplo es la etiqueta < video/> en HTML5.
* **Te permite usar el Shadow DOM (Shady DOM):** Hasta ahora, el uso del Shadow DOM estaba reservado para los creadores de los browsers. Esta es la tecnología que permite encapsular el contenido de un Custom Element del resto. Y así evita que sus estilos y código interactúen de forma no deseada con el resto del sitio[.](http://www.w3schools.com/html/html5_video.asp) Polymer 1.0 incluye una tecnología llamada Shady DOM que te permite desarrollar Custom Elements seguros y eficientes encapsulados en el Shadow Dom.
* **Habilita HTML Imports en tu aplicación:** HTML Imports te permite desarrollar aplicaciones web modulares de forma sencilla y eficiente al permitirte colocar todo lo relacionado con una parte de tu aplicación (por ejemplo, un Custom Element) en un documento HTML. Por ejemplo, puedes colocar todas las librerías que usa tu aplicación en un archivo llamado librerías.html y las hojas de estilo en uno llamado estilos.html. Después, en tu aplicación sólo debes importar estos archivos de este modo:

<link href="librerias.html" rel="import">

<link href="estilos.html" rel="import">

Así podemos crear código mucho más fácil de leer. Además, el tener todo por separado facilita el realizar ajustes más adelante.

* **Polymer 1.0 viene listo para producción:** Está nueva versión está optimizada e incluye todo lo necesario para usarlo de forma exitosa en una aplicación real. No sólo garantiza estabilidad y rendimiento; sino que también incluye un conjunto de herramientas y tareas necesarias para usarlo profesionalmente en el Polymer Starter Kit.
* **Te permite desarrollar aplicaciones confiables con Unit Testing:** Polymer incluye todo lo necesario para realizar Unit Testing en tus Web Components con una herramienta llamada Web-Component-Tester. Esta se encarga de comprobar desde la consola elfuncionamiento correcto de tus componentes en los principales navegadores. Una vez que tu componente pasa las pruebas, puedes usarlo con total seguridad.
* **Incluye entorno de desarrollo, producción y un servidor de archivos:** Polymer Starter Kit viene configurado para trabajar cómodamente en un entorno de desarrollo con todos sus componentes separados. De esta manera, lograrás trabajar de forma eficiente. Además, compilará todo esto en archivos listos para producción, que colocará en una carpeta especial. Y, además, incluye un servidor de datos. Cuando se usa como servidor local, este actualiza el navegador de forma automática cuando realizas un cambio en el código. Y cuando se encuentra en modo servidor remoto, ocupa una versión optimizada de tu aplicación y un caché con los archivos principales.
* **Licencias:**
* **BSD-old:** Esta versión ha sido examinada como una licencia de código abierto por OSI como "The BSD License".

La Free Software Foundation, que se refiere a la licencia como la "Licencia BSD modificada", afirma que es compatible con la GNU GPL. La FSF alienta a los usuarios a ser específicos al referirse a la licencia por su nombre (es decir, no simplemente hacer referencia a ella como "licencia BSD" o "estilo BSD") para evitar confusiones con la licencia BSD original.



Figura 4 Licencia.

* **METEOR JS**
* **Definición:**

Meteor es una plataforma de open source para la web, móvil y escritorio. Es un framework para aplicaciones web con JavaScript.

Meteor funciona con un rendereo reactivo en donde se integra Angular JS y React JS o el framework Blaze. Así como también cuenta con una plataforma online, usa menos código y crea interfaces de usuario eficientes.

Meteor, un medio ambiente simple que permite desarrollar aplicaciones en poco tiempo.

* **Ventajas de uso:**
* **Las aplicaciones son en tiempo real.**

Meteor cuenta en su núcleo con instrucciones necesarias para hacer cualquier cambio realizado en la aplicación web y así mismo este actualice la base de datos y traiga el refresco de la plantilla de visualización.

* **Todo es con un único lenguaje.**

Meteor simplifica el proceso al reducir el alcance de lo que se debe de tener en cuenta. Gracias a que se emplea un único lenguaje es más fácil para el desarrollador trabajar en el frontend, backend y así mismo la base de datos.

* **Smart Packages:**

Meteor ofrece paquetes de código ya escrito que aportan funciones adicionales al proyecto.

* **Soporte:**

Existe muchísima documentación al respecto mediante una comunidad y foros, en donde se detalla cualquier proceso a seguir.

* **Ideal para principiantes:**

Se comienza un proyecto muy rápido más idóneo y sencillo para desarrollar.

* **Tecnologías que emplea:**
* Node js
* Npm
* Java script
* Angular js
* React js
* Blade
* Vue js
* Mongo db
* **Librerías que usa:**
* **Meteor js:** Meteor proporciona varias ventajas para el usuario además de su reactividad.  
  Imaginen que Meteor JS brinda dos bases de datos:
* Una base de datos del lado del cliente (cache).
* Una base de datos MongoDB en el servidor.

Cuando un usuario modifica algún dato—haciendo clic en Save, por ejemplo, el código JavaScript que se ejecuta en el navegador actualiza la base de datos en MongoDB local y luego realiza una solicitud de DDP al servidor. El código actúa de inmediato como si la operación hubiera sido exitosa ya que no necesita esperar la respuesta por el servidor.

Mientras tanto, el servidor se actualiza en segundo plano. Si se produce un error en la operación del servidor o si devuelve un resultado inesperado, el código JavaScript del lado del cliente se reajusta inmediatamente según los datos devueltos recientemente del servidor. Este ajuste se denomina compensación de latencia y genera una cantidad de velocidad adicional que el usuario percibe.

* **Licencias:**
* **MIT:** La licencia MIT es una de tantas licencias de software que ha empleado el Instituto Tecnológico de Massachusetts.

El Instituto Tecnológico de Massachusetts fue fundado en 1861 por el geólogo William Barton Rogers en la zona de Cambridge, en Massachusetts. Surgió como respuesta a la creciente industrialización del país y, desde sus inicios, se configuró como un modelo europeo de universidad politécnica dedicada al estudio de las ciencias físicas e ingenierías para contribuir al desarrollo industrial.

* **Características y usos de esta licencia**

Esta licencia permite reutilizar el software así licenciado tanto para ser software libre como para ser software no libre, permitiendo no liberar los cambios realizados al programa original.

También permite licenciar dichos cambios con licencia BSD, GPL u otra cualquiera que sea compatible (es decir, que cumpla las cláusulas de distribución). Con esta licencia se tiene software libre. Ejemplos en los que podría interesar su aplicación serían las licencias duales (El uso de una licencia dual, de doble licenciamiento o de múltiple licenciamiento es la práctica de conceder dos o más licencias para un mismo producto con el objetivo de compatibilizar licencias o segregar segmentos del mercado). Si se pretende difundir un estándar mediante una implementación de referencia, o si simplemente se pretende que el producto sea Libre sin mayores consideraciones.

* **Licencia BSD**

Esta licencia tiene menos restricciones en comparación con otras como la GPL estando muy cercana al dominio público. La licencia BSD al contrario que la GPL permite el uso del código fuente en software no libre. Es una licencia de software libre permisiva como la licencia MIT. Esto está en contraste con las licencias copyleft, que tienen de reciprocidad requisitos de compartir-igual.

* **Licencia GPL**

Es una licencia de derecho de autor ampliamente usada en el mundo del software libre y código abierto, ​y garantiza a los usuarios finales (personas, organizaciones, compañías) la libertad de usar, estudiar, compartir (copiar) y modificar el software.

* **MOTION UI**
* **Definición:**

Motion UI es una biblioteca Sass para crear rápidamente transiciones y animaciones de CSS.

Motion UI incluye un conjunto de efectos prefabricados como clases de CSS. Esto incluye los efectos de transición a deslizamiento, fundido, bisagra, escala y giro, así como algunas animaciones incorporadas.

La función central de Motion UI es la transición de componentes hacia adentro y hacia afuera.

* **Desarrollada por zurb:**

Son un equipo de diseñadores, investigadores y nerds que ayuda a las empresas a diseñar mejores productos, sitios web y servicios mediante el descubrimiento de #DesignInsights (Información sobre el diseño).

* **Tecnologías que emplea:**

El paquete Motion UI incluye una pequeña biblioteca de JavaScript para activar estas transiciones.

* **JavaScript:** JavaScript es un lenguaje de programación dinámica multiparadigma de alto nivel de HTML y la Web que se utiliza para mejorar la interactividad en los sitios web. A menudo abreviado simplemente como JS, JavaScript es un lenguaje de scripting basado en prototipos que se utiliza dentro de los navegadores web para interacciones de interfaz tales como respuestas, juegos, animación, estilo dinámico, etc. Es un lenguaje versátil pero sofisticado utilizado para la programación en la web. JavaScript es una de las principales tecnologías de producción de contenidos de la World Wide Web, junto con HTML y CSS. Es muy flexible con funciones de primera clase. Aunque es un lenguaje de scripting, también se puede utilizar para entornos que no sean navegadores.
* **JQuery:** jQuery es una librería de JavaScript eficaz y rápida que simplifica el desplazamiento de documentos HTML, la animación, el manejo de eventos y la interacción para Ajax. Es una librería de scripting multiplataforma rica en funcionalidades especialmente diseñada para manejar scripting del lado del cliente de HTML. La sintaxis de jQuery está diseñada para hacer las cosas simples como animaciones, elementos HTML, oyentes de eventos, etc. Las cosas son mucho más fáciles con jQuery que es muy ligero y rápido, y soporta todo tipo de navegadores web. La idea es facilitar a los desarrolladores el uso de JavaScript en sitios web para que sea más interactivo y fácil de usar con menos codificación, por supuesto.
* **SASS:** (Syntactically Awesome Style Sheets) es el lenguaje de extensión CSS de grado profesional más maduro, estable y poderoso del mundo. Es el mejor ayudante para el desarrollador web a la hora de trabajar en CSS gracias a sus múltiples funcionalidades: mixins, variables, funciones, herencia, nesting.

La principal ventaja de SASS es la posibilidad de convertir los CSS en algo dinámico. Permite trabajar mucho más rápido en la creación de código con la posibilidad de crear funciones que realicen ciertas operaciones matemáticas y reutilizar código gracias a los mixins, variables que nos permiten guardar valores.

SASS dispone de dos formatos diferentes para la sintaxis, lo que hace se traduce en dos extensiones de fichero diferentes: .sass y .scss

* **Ventajas de uso:**
* Se puede adaptar fácilmente para trabajar con cualquier biblioteca de animación de frameworks, como Angular o React.
* Motion es una herramienta poderosa para los diseñadores, que nos permite diseñar interfaces más expresivas al darles a los usuarios colas alrededor de la jerarquía, la orientación espacial, y más.
* **Licencias:**
* **Licencia MIT**

La licencia MIT es una de tantas licencias de software que ha empleado el Instituto Tecnológico de Massachusetts (Massachusetts Institute of Technology) a lo largo de su historia.

Esta licencia es una Licencia de software libre permisiva lo que significa que impone muy pocas limitaciones en la reutilización.

La licencia MIT siempre ha sido una licencia importante y se utiliza muy a menudo en el Software libre. En 2015, de acuerdo con Black Duck Software y datos de GitHub, ​ se convirtió en la licencia más popular por encima de las variantes de la licencia GPL.

* El texto diferencia tres puntos:
* **Condiciones**, la condición es que la nota de copyright y la parte de los derechos se incluya en todas las copias o partes sustanciales del Software. Esta es la condición que invalidaría la licencia en caso de no cumplirse.
* **Derechos**, los derechos son muchos: sin restricciones; incluyendo usar, copiar, modificar, integrar con otro Software, publicar, sublicenciar o vender copias del Software, y además permitir a las personas a las que se les entregue el Software hacer lo mismo.
* **Limitación de responsabilidad**, finalmente se tiene una nota de limitación de la responsabilidad habitual en este tipo de licencias.
* **APLICACIONES WEB PROGRESIVAS (PWA)**
* **Definición:**

Las PWA son sitios web avanzados y deben cumplir con las siguientes características:

* **Progresiva**: funciona para todos los usuarios, sin importar la elección de navegador, porque está construida con mejora progresiva como principio central.
* **Adaptable**: se adapta a cualquier factor de formulario, sea escritorio, móvil, tablet o lo que venga en el futuro.
* **Independiente de la conectividad**: mejorada con service workers para trabajar sin conexión o con redes de mala calidad.
* **Estilo app**: al usuario le parece una app con interacciones y navegación estilo app, porque está construida con modelo de shell de app.
* **Fresca**: siempre actualizada gracias al proceso de actualización de service worker.
* **Segura**: emitida vía HTTPS para evitar intromisiones y para garantizar que el contenido no se haya manipulado.
* **Descubrible**: se puede identificar como "app" gracias al manifiesto W3C y al alcance de registro de service worker, lo que permite que los motores de búsqueda la encuentren.
* **Posibilidad de volver a interactuar**: facilita la posibilidad de volver a interactuar a través de funciones como notificaciones push.
* **Instalable**: les permite a los usuarios "conservar" las apps que les resultan más útiles en su pantalla principal sin la molestia de una tienda de app.
* **Vinculable**: se puede compartir fácilmente vía URL, no requiere instalación compleja.

Una PWA utiliza las últimas tecnologías disponibles en los navegadores para ofrecer una experiencia en móviles lo más parecida a la de una aplicación nativa.

* **Tecnologías que emplea:**
* Responsive Web Design, animaciones CSS y frameworks específicos para crear interfaces móviles con aspecto de nativas.
* **Service Workers:** Van más allá de lo que ofrece un Web Worker. Éstos últimos nos permiten ejecutar código pesado en segundo plano (en un subproceso dedicado) y comunicarnos con ellos, de modo que una o varias tareas largas no bloqueen la interfaz de usuario. Pero los Service Workers son más potentes y complejos, puesto que pueden ejecutarse de manera independiente a la aplicación (es decir, estar en ejecución, aunque la página de nuestra app web esté cerrada) y ofrecen capacidades avanzadas como la intercepción de las comunicaciones, el cacheado de información, la descarga en segundo plano de contenidos, el trabajo sin conexión o la posibilidad de enviar notificaciones.

En realidad, para algunas cosas no es necesario utilizar un Service Worker. Por ejemplo, es posible crear aplicaciones web que funcionen off-line (sin conexión) utilizando la API de AppCache. El Service Worker nos puede dar más funcionalidad, sobre todo si necesitamos tomar decisiones a la hora de cachear la información y no solo asegurar que se encuentra almacenada, pero no son estrictamente necesarios. Es decir, para crear una PWA no es necesario estrictamente usar Service Workers salvo que requiramos ciertas funcionalidades avanzadas.

* **App Shell:**
* App Shell no es una tecnología, sino un modelo o patrón a la hora de crear las aplicaciones. La idea es muy sencilla: separar la aplicación entre funcionalidad y contenido y cargarlos por separado.
* Lo cierto es que la mayor parte de las aplicaciones de tipo Single Page (SPAs) ya suelen usar una u otra forma de hacer eso.
* Lo suyo es tener, por un lado, la aplicación en sí cacheada para uso off-line (con Service Workers o no) de modo que cargue a toda velocidad, y luego el contenido (los datos) que cargue por otro lado, bien de una caché inicial también y luego se actualicen, o directamente desde la web si hay conexión.
* Esto, bien realizado, hace que la percepción que tiene el usuario de la velocidad de carga de la app sea mayor. Parece que carga mucho más rápido porque al cargar el "shell" antes de nada y desde una caché, el usuario verá la app enseguida.
* **Manifiesto de aplicación:**
* Como comentaba al principio, desde los primeros *smartphones* como hoy los conocemos, siempre ha sido posible anclar al inicio una página web desde el navegador para luego poder ir directamente a ella. Para controlar el aspecto que tendrá el icono que los usuarios van a anclar es posible utilizar diversas técnicas dependiendo del navegador y el sistema operativo. Así, en iOS o Windows Phone eso se controla a través de unas cabeceras de tipo "meta" que podemos añadir a la página principal de la aplicación web. En el caso de Android y Chrome se utiliza un archivo llamado "Manifiesto" cuyo nombre es manifest.json (que funciona hace años, desde la versión 38 de Chrome en 2014).
* Hace poco Google ha hecho que cuando se añade una aplicación al menú de inicio de Android salga un banner de instalación como el de una aplicación real, todo ello conducente a que la experiencia cada vez sea más parecida a la de las aplicaciones nativas. Pero no deja de ser una cuestión cosmética.
* Esto, nuevamente, solo funciona con Chrome en Android, así que no sirve de mucho en otros sistemas, aunque todos tienen algo similar para definir el aspecto de los accesos directos.
* **Ventajas de uso:**
* Combinan los beneficios de una aplicación web con la experiencia de una app nativa.
* Pueden ser linkeables para compartir con demás usuarios sin importar desde donde o como entres a la app.
* No es necesario instalarlas.
* Cargan inmediatamente.
* Es posible agregar un acceso directo a nuestro móvil.
* Combina lo mejor de las aplicaciones web y las aplicaciones móviles nativas.
* Posibilidad de ingresar el hardware del navegador.
* **Licencias:**

La mayoría de las tecnologías utilizadas para desarrollar progresive web apps son de licencia open source.

* **IONIC**
* **Definición:**

Ionic 2 es un framework para el desarrollo de aplicaciones híbridas, inicialmente pensado para móviles y tablets, aunque ahora también capaz de implementar aplicaciones web e incluso dentro de pocas aplicaciones de escritorio multiplataforma. Su característica fundamental es que usa por debajo Angular 2 y una cantidad de componentes enorme, que facilita mucho el desarrollo de las aplicaciones.

Se trata de una estupenda herramienta para la creación de aplicaciones sorprendentes, pensada para obtener resultados de una manera rápida y con una menor inversión económica, ya que permite crear aplicaciones para distintas plataformas móviles con una misma base de código.

Este es el primero de una serie de artículos para explicar Ionic 2 con el que iremos componiendo el Manual de Ionic 2 de DesarrolloWeb.com. Lo estamos desarrollando ya con la versión final de Ionic 2, la versión más nueva del framework ya estable y preparada para producción. Aunque seguro que estás deseando empezar por instalar Ionic y comenzar tu primer proyecto, en este momento comenzaremos por aclarar algunos conceptos.

* **Tecnologías que emplea:**
* Node js
* Cordova
* Angular js.
* Npm
* Sass
* **Ventajas:**
* **Utilidad Multiplataforma:** Ionic es compatible en varias plataformas y puede ser optimizado para varios sistemas operativos móviles. Desarrollar código es mucho más fácil y rápido ya que Ionic se integra fácilmente con Angular JS para configurar la estructura del código. Esto lleva al desarrollo de nuevos conceptos, más rápido y con presupuestos más bajos.
* **Interfaz de usuario Amigable:** Ionic incorpora conceptos clave como Java Scripts y componentes CSS para optimizar interfaces fáciles de usar en cualquier aplicación móvil. Componentes como pestañas de navegación, menú deslizable, botones, listas, entradas de formularios, ventanas emergentes y avisos son muy simples y elegantes y fácilmente personalizables para una aplicación móvil concreta.

Ionic, por lo tanto, crea interfaces de usuario atractivas e interactivas en un período de tiempo notablemente corto.

* **Framework Libre y de Código Abierto:** Dado que Ionic es de código abierto, los desarrolladores pueden personalizar distintos diseños en varios sistemas operativos móviles sin incurrir en grandes costos. Ionic da códigos de componentes CSS, JS y HTML que reducen la necesidad de reescribirlo para un nuevo sistema operativo móvil. La fácil integración de Ionic con Angular JS facilita la creación de una estructura de código más fácil y mejor.

Desarrollar aplicaciones en Windows, Android e iOS es así más rápido y fácil de hacer. La creacion de Apps pasan rápidamente del desarrollo al mercado y esto aumenta la comercialización y crea conciencia y popularidad de las aplicaciones. Ionic ayuda a ahorrar dinero, tiempo y esfuerzos.

* **Desarrollo Posible y Fácil de Aplicaciones de Cross Mobile:**

Desarrollar una aplicación rápidamente es fundamental en la generación de aplicaciones móviles actuales. El otro concepto es que el desarrollo de aplicaciones debe ser compatible con todos los dispositivos móviles.

Ionic se ha destacado en el desarrollo de aplicaciones de manera eficiente, utilizando herramientas estándar con una única base de codificación que ahorra dinero, tiempo y esfuerzo y proporciona una apariencia integrada.

* **Basado en Angular:** Ionic se basa en la compatibilidad con Angular JS que luego amplía la facilidad y funcionalidad del framework de Angular para hacer la creación de aplicaciones móviles muy fáciles. A partir de hoy, Angular JS nos muestra el marco favorito de JavaScript en uso, que cuenta con el respaldo de Google.

* **Licencias**
* **MIT**

La licencia MIT es una de tantas licencias de software que ha empleado el Instituto Tecnológico de Massachusetts (Massachusetts Institute of Technology) a lo largo de su historia.

Esta licencia es una Licencia de software libre permisiva lo que significa que impone muy pocas limitaciones en la reutilización.

La licencia MIT siempre ha sido una licencia importante y se utiliza muy a menudo en el Software libre. En 2015, de acuerdo con Black Duck Software y datos de GitHub, ​ se convirtió en la licencia más popular por encima de las variantes de la licencia GPL.

“Se concede permiso, de forma gratuita, a cualquier persona que obtenga una copia de este software y los archivos de documentación asociados (el "Software"), para tratar el Software sin restricciones, incluyendo sin limitación, los derechos de uso, copia, modificación, fusión, publicación, distribuir, sublicenciar y / o vender copias del Software, y Permitir a las personas a quienes se les proporciona el Software hacerlo, sujeto a las siguientes condiciones:

El aviso de copyright anterior y este aviso de permiso serán incluido en todas las copias o porciones sustanciales del software.

El software se proporciona "tal cual", sin garantía de ningún tipo, expreso o implicado, incluidas, entre otras, las garantías de comerciabilidad, aptitud para un propósito particular y no infracción en ningún caso serán los autores o los titulares de los derechos de autor responsable por cualquier reclamo, daños u otra responsabilidad, ya sea en acción del contrato, extracontractual o de otro tipo, derivados, fuera de o en conexión con el software o el uso u otros tratos en el software.”

* **GPL**

Los programas de ordenador suelen distribuirse con licencias propietarias o cerradas. Estas licencias son intransferibles y no exclusivas, es decir, no eres propietario del programa, sólo tienes derecho a usarlo en un ordenador o tantos como permita expresamente la licencia y no puedes modificar el programa ni distribuirlo.

La licencia GPL o General Public License, desarrollada por la FSF o Free Software Foundation, es completamente diferente. Puedes instalar y usar un programa GPL en un ordenador o en tantos como te apetezca, sin limitación. También puedes modificar el programa para adaptarlo a lo que tú quieras que haga. Además, podrás distribuir el programa GPL tal cual o después de haberlo modificado.

Puedes hacer esto, regalando el programa o vendiéndolo, tu única obligación, es facilitar siempre con el programa binario el código fuente, es decir, el programa de forma que pueda ser leido por un programador.Los programas propietarios o cerrados, solo se distribuyen en binario, listos para ejecutarse en el ordenador.

Un programa GPL es libre, es decir, no es \"shareware\" o \"freeware\", éstos también puedes conseguirlos en un cd-rom al comprar una revista o descargarlos de internet, pero no incluyen el código fuente y no tienes permiso para modificarlos, es decir, no son libres.

Los programas distribuidos con licencia GPL tampoco son piratas, utiliza otra licencia para atacar barcos. Los programas GPL no tienen garantía, igual que casi todos los programas propietarios, no obstante, ofrecen más derechos a sus usuarios y su sistema abierto hace que los defectos sean detectados y depurados a gran velocidad con la ayuda de cientos de programadores a través de internet. Por otro lado, nada impide a una empresa garantizar el Software Libre junto a otros servicios que oferte.

* **OPEN SOURCE**

Una licencia open-source es una licencia que se usa para programas de computadoras, con copyright, que siguen los principios del movimiento Open Source.

Más formalmente, una licencia es considerada Open Source cuando ha sido aprobada por la Open Source Initiative (OSI), donde el criterio lo da la definición de Open Source. El software de dominio público (esto significa sin licencia), cumple todos estos criterios siempre y cuando todo el código fuente esté disponible, y esté reconocido por la OSI y se le permita usar la marca de esta.

* **Definición de Open Source**

La Open Source Initiative utiliza la Definición de Open Source para determinar si una licencia de software de computadora puede o no considerarse software abierto. La definición se basó en las Directrices de software libre de Debian, fue escrita y adaptada primeramente por Bruce Perens. Es similar pero no igual a la definición de licencia de software libre.

Bajo la Definición Open Source, las licencias deben cumplir diez condiciones para ser consideradas licencias de software abierto:

* Libre redistribución: el software debe poder ser regalado o vendido libremente.
* Código fuente: el código fuente debe estar incluido u obtenerse libremente.
* Trabajos derivados: la redistribución de modificaciones debe estar permitida.
* Integridad del código fuente del autor: las licencias pueden requerir que las modificaciones sean redistribuidas solo como parches.
* Sin discriminación de personas o grupos: nadie puede dejarse fuera.
* Sin discriminación de áreas de iniciativa: los usuarios comerciales no pueden ser excluidos.
* Distribución de la licencia: deben aplicarse los mismos derechos a todo el que reciba el programa.
* La licencia no debe ser específica de un producto: el programa no puede licenciarse solo como parte de una distribución mayor.
* La licencia no debe restringir otro software: la licencia no puede obligar a que algún otro software que sea distribuido con el software abierto deba también ser de código abierto.
* La licencia debe ser tecnológicamente neutral: no debe requerirse la aceptación de la licencia por medio de un acceso por clic de ratón o de otra forma específica del medio de soporte del software.

## MEDIOS DIGITALES

### Audio

* **AAC:** Es un formato de compresión de datos de audio desarrollado por el Instituto Fraunhofer conjuntamente con AT&T, Nokia, Sony y Dolby.
* **Características:** El algoritmo que se utiliza tiene un rendimiento superior al del MP3, que produce una mejor calidad en archivos pequeños y requiere menos recursos del sistema para codificar y decodificar.
* **Ventajas:**
* Puede un soportar uno sonido multicanal incluso para más de seis altavoces en modo envolvente.
* La calidad de un archivo AAC a 64kbps puede ser la misma de un mp3 a 128 kbps logrando el primero formato obtener menos espacio.
* Puede tener menos pérdida de sonido a bitrates bastante reducidos.
* **Desventajas:**
* Las personas NO están más adaptadas con este formato de Audio.
* **MP3 o MPEG-3**

Es una forma de codificar audio usando un algoritmo de compresión que genera pérdida de datos reduciendo la cantidad de información requerida para representar una grabación de audio, pero de modo en que a su vez sea muy similar a la calidad del archivo original para la mayoría de los oyentes.

* **Características:**
* Este formato permite reducir el número de bytes en una canción sin dañar la calidad del sonido.
* La meta del formato MP3 es comprimir una canción de calidad CD sin que apenas se aprecie la diferencia.
* Con MP3, una canción de 32 MB de un CD, se comprime hasta 3 MB. Esto te permite descargarte una canción en minutos en lugar de horas, y almacenar cientos de canciones en el disco duro de tu ordenador.
* **Ventajas:**
* Soportado por casi todos los programas reproductores o reproductores de audio portátiles.
* ISO estándar, parte de especificación MPEG.
* Decodificación rápida.
* Cada uno puede crear su encoder/decoder.
* Utiliza menos espacio que la canción original, o el archivo original.
* **Desventajas:**
* La calidad del sonido no es buena.
* El volumen a veces sube y baja y suena a estática.
* Pérdida de calidad en todo el formato.
* Pierden información del audio, lo que causa la mala calidad de sonido.
* Se pierden los derechos de autor.
* **FLAC**

Este formato de audio **comprime los archivos para disminuir su tamaño sin perder calidad.** Un punto importante del formato es que es gratuito, de código abierto y con una licencia libre de regalías.

* **Características:**
* **Losseless:** no hay pérdida de información en el proceso de codificación y el audio decodificado es idéntico al utilizado en la codificación. Cada cuadro contiene un CRC de 16 bits de la información para la detección de errores de transmisión. La integridad de la información de audio es asegurada almacenando una firma MD5 de la información del audio original en la cabecera del archivo, con la que se puede comparar durante la decodificación.
* **Rapidez:** FLAC está concebido para la velocidad de decodificación que es mucho menos intensiva en el cómputo que otros códecs. EL rendimiento de la decodificación en tiempo real se logra fácilmente en hardware modesto.
* **Flexibilidad de metadatos**: el sistema de metadatos soporta etiquetas, portada de álbum, seek tables y cue sheets. Las aplicaciones pueden escribir sus propios metadatos. Nuevos bloques de metadatos se pueden definir e implementar en versiones futuras sin romper decodificadores viejos.
* **Seekable:** soporta la búsqueda rápida muy útil en la reproducción y en aplicaciones para la edición de audio.
* **Capacidad para streaming:** cada cuadro contiene la información suficiente para decodificar ese cuadro
* **Resistencia a errores:** además de las características anteriores, el códec tiene la particularidad de no contar con métodos de prevención de copia.
* **Ventajas:**
* Es un formato de audio sin perdidas libre.
* La calidad es igual a la del APE.
* Utilizamos menos procesador.
* Es multicanal (acepta todo streaming).
* Evita molestos cortos.
* Es muy compatible.
* **Desventajas:**
* La mayoría de los dispositivos externos de mp3 no lo soportan todavía de forma nativa.
* **OGG VORBIS**

Es una forma de compresión que reduce algunos de los datos de audio y se denomina "compresión con pérdida". (lossy compression) Esto quiere decir que se eliminan algunos de los datos de audio (frecuencias inaudibles, por ejemplo), a fin de obtener el mayor grado de compresión posible y generar un archivo de salida que suene lo más parecido al original.

* **Características:**
* Es un códec de audio libre que permite una máxima flexibilidad a la hora de elegir entre la amplia gama de bitrates según la complejidad de la transmisión de audio.
* En la relación calidad-bitrate, se encuentra parejo con MPEG-2 y en la mayoría de los bitrates es comparable al formato ACC.
* **Ventajas:**
* Está disponible para la comunidad, que puede seguir mejorándolo.
* Tiene excelentes tasas de compresión.
* Añade cierto brillo artificial en las frecuencias altas, siendo muy inferiores las pérdidas producidas por la teoría psicoacustica.
* Permite almacenar más canales de audio.
* El contenedor .ogg permite almacenar audio y vídeo.
* A igualdad de tamaño de archivo la calidad del formato ogg es superior.
* La calidad de un archivo .ogg puede ser muy superior a la de un mp3.
* Ogg Vorbis es abierto y libre de patentes
* El sistema de etiquetado es mejor en ogg
* **Desventajas:**
* No es compatible con MP3. Esto quiere decir que el usuario tiene que utilizar un reproductor de audio que soporte el formato o instalar un códec específico para reproducir los archivos
* El consumo de batería es superior.
* **WAV**

Es un formato de audio digital sin compresión que se emplea para almacenar sonidos para ordenadores con Windows, es un formato parecido al AIFF, pero tomando en cuenta peculiaridades de Intel.

* **Características:**
* Es ideal para guardar audios originales a partir de los cuales se puede comprimir y guardar en distintos tamaños de muestreo para publicar en la web.
* Es un formato de excelente calidad de audio.
* Compresión: Los archivos WAV se pueden guardar con distintos tipos de compresión. Las más utilizadas son la compresión PCM y la compresión ADPCM.
* Se suele utilizar para fragmentos muy cortos (no superiores a 3-4 segundos), normalmente en calidad mono y con una compresión Microsoft ADPCM 4 bits.
* **Ventajas:**
* Puede contener otras resoluciones o incluso ser monofónicos.
* Se puede considerar mucho para grabar voces.
* **Desventajas:**
* Se considera insuficiente para la música.
* Las más comunes son a 16 bits y 44,1kHz.
* Las WAV monofónicos a 8 bits y 22kHz eran solo capaces de reproducir esta resolución.
* Produce archivos de un peso enorme. Una canción extraída de un CD (16 bytes, 44100 Hz y estéreo) puede ocupar entre 20 y 30 Mb.

### Video

Es la tecnología de la captación, grabación, procesamiento, almacenamiento, transmisión y reconstrucción por medios electrónicos digitales o analógicos de una secuencia de imágenes que representan escenas en movimiento. (EcuRed, s.f.)

* **WEBm**

WebM es un formato multimedia abierto y libre desarrollado por [Google](https://es.wikipedia.org/wiki/Google) y orientado a usarse con [HTML5](https://es.wikipedia.org/wiki/HTML5).

Está compuesto por el [códec](https://es.wikipedia.org/wiki/C%C3%B3dec) de vídeo [VP8](https://es.wikipedia.org/wiki/VP8) (desarrollado originalmente por [On2 Technologies](https://es.wikipedia.org/wiki/On2_Technologies)) y el códec de audio [Vorbis](https://es.wikipedia.org/wiki/Vorbis) dentro de un contenedor multimedia [Matroska](https://es.wikipedia.org/wiki/Matroska" \o "Matroska).

* **Características:**
* Los navegadores web Mozilla Firefox, Opera y Google Chrome soportan la reproducción de archivos de vídeo WEBM, pero Internet Explorer 9 y Safari requieren la asistencia de terceros (como QuickTime) para reproducir medios WEBM.
* Menor peso y misma calidad que otros formatos de video.
* **Ventajas:**
* Es un formato pensado para su inserción con la etiqueta vídeo de HTML5.
* Al estar pensado para la web, puedes obtener casi la misma calidad que un MP4 con menos de la mitad de tamaño.
* Funciona muy bien como Streaming.
* **Desventajas:**
* No es compatible con todos los navegadores.
* No hay cámaras que graben en formato WEBm de forma nativa, por lo que será necesario usar convertidores.
* **MP4**

Es un formato de codificación de audio asociado a la extensión mp4. MPEG4 es un códec estándar internacional de vídeo creado especialmente para la web. Es un algoritmo de compresión que codifica datos audio vídeo optimizando su calidad de almacenamiento, codificación y distribución en redes.

* **Características:**
* Compatible con la mayoría de los navegadores web o con extensiones de terceros.
* MP4 permite streaming de contenido bajo demanda vía [Internet](https://www.ecured.cu/Internet), así como el multiplexado de múltiples pistas de audio y video en un único archivo, bitrates y cuadros por segundo variables, subtítulos, etc.
* Permite un menor tiempo de descarga.
* **Ventajas:**
* El formato de archivo MP4 da una impresión de calidad muy parecida a la calidad de vídeo del DVD original. Y los vídeos de este tipo se pueden descargar desde los sitios web de forma rápida y sencilla con la conexión de banda ancha de alta velocidad disponible.
* MP4 permite la transmisión a través de Internet, lo que indica que el contenido se puede ver en tiempo real.
* MP4 admite hardware de gran tamaño. Como un formato de vídeo estándar, dispositivos y aparatos digitales modernos se pueden reproducir sin problemas.
* La tecnología de MP4 es ampliable. Se puede adaptar para crear la base para el desarrollo de nuevas tecnologías incluidas en la “familia MP4”.
* **Desventajas:**
* Mp4 admite solo una pista de audio.
* **AVI**

Los archivos AVI ([Audio](http://culturacion.com/como-escoger-una-tarjeta-de-sonido/) Video Interlieve), son conocidos también como archivos de audio y video intercalados. Estos fueron creados por Microsoft, para contener al mismo tiempo archivos de audio y video.

* **Características:**
* Posibilidad de contener múltiples códecs que proporcionarán ventajas a nuestra web.
* Cuenta con la mayoría de códecs de audio.
* **Ventajas:**
* La principal ventaja es que, actualmente, casi todos los reproductores multimedia cuentan con los codecs necesarios para reproducir este tipo de archivo, por lo que se considera un archivo de tipo universal.
* Los archivos AVI pueden contener varios archivos de audio para un mismo archivo de video, ofreciéndole al usuario la posibilidad (en muchos casos), de escoger la calidad de audio que prefiera para disfrutar de sus archivos multimedia.
* Los anuncios web, promociones y películas cortas generalmente recurren a formatos de archivo AVI debido a su mayor capacidad de compresión.
* **Desventajas:**
* Comprimir archivos AVI más allá de cierto grado puede resultar en mala calidad o pérdida de calidad del video.
* Los archivos AVI pueden ser formatos de archivo muy grandes si no los comprime.
* No es muy utilizado en streamings web debido a su peso.

### Imagen

Imagen digital. Es un producto del desarrollo de la informática que tiene como antecesor a la fotografía. La imagen digital toma vida mediante un archivo de diferentes formatos, que puede ser almacenado en una PC.

Las imágenes digitales se dividen en dos tipos: imágenes vectoriales y de mapa de bits.

Las imágenes de mapa de bits son imágenes pixeladas, es decir que están formadas por un conjunto de puntos (píxeles) contenidos en una tabla. Cada uno de estos puntos tiene un valor o más que describe su color.

Las imágenes vectoriales son representaciones de entidades geométricas tales como círculos, rectángulos o segmentos. Están representadas por fórmulas. El procesador traducirá estas formas en información que la tarjeta gráfica pueda interpretar. (EcuRed, s.f.)

* **JPEG / JPG**

Es de lo más usados, sobre todo en internet, en fotografías o en gráficos web. Es un formato de imagen con pérdidas y pesa poco, aunque en resoluciones pequeñas puede aumentar notablemente su pixelación.

Se recomienda usar el formato de imagen para imprimir archivos en alta resolución (pero con una baja compresión para evitar demasiadas pérdidas).

* **Características:** Permite imágenes de hasta 16 millones de colores, permite una gran compresión con bastante buena calidad. Además, es un formato universalmente aceptado, se puede ver desde cualquier navegador web y desde cualquier herramienta gráfica.
* **Ventajas:**
* Es uno de los formatos más comunes para la distribución de la fotografía y otros gráficos en la Web.
* La mayoría de las cámaras digitales de captura de imágenes en formato JPG por defecto.
* Es capaz de crear archivos de pequeño tamaño.
* Muy alto grado de compresión.
* **Desventajas:**
* La edición de imágenes se ve dificultada.
* La calidad disminuye cada vez que se guarda el archivo.
* No permite transparencias ni animación.
* **GIF**

Es un formato de imagen sin pérdidas y se usa casi exclusivamente para web ya que permite crear pequeñas animaciones.

Se recomienda usar el formato de imagen GIF para animaciones web como banners, memes o iconos. Cabe mencionar que puedes guardarlo con un fondo transparente.

* **Características:**
* **Soportan Animaciones:** Genera gráficos con movimiento.
* **Soportan Transparencias:** Esto hace que se muestre el fondo sobre el que se apoya la imagen.
* **Efecto de difusionado o entrelazado:** Funciona de manera que conforme se va cargando va mostrando la imagen más nítidamente, hasta alcanzar la resolución total de la misma.
* **Solo soporta 256 colores:** Se emplea en dibujos, no se recomienda este formato para fotografías ya que estas requieren muchos colores.
* **Comprime las imágenes:** Esto es muy útil para empleo de Internet, ya que la velocidad de transmisión es muy lenta para imágenes, sonido y vídeo, ya que estos suelen ser archivos grandes.
* Es aceptado por la mayoría de los navegadores y herramientas gráficas.
* **Ventajas:**
* Últimamente ha adquirido gran popularidad gracias a los memes.
* Sirve para realizar efectos de aclaramientos, transparencias y animaciones.
* Formato de compresión sin perdida.
* Ideal para páginas web con gráficos pequeños.
* El peso de los archivos es considerablemente menor al de un video, por lo que subirlo o descargarlo no consume mucho. Al mismo tiempo, algunas aplicaciones lo han incorporado como acceso rápido para ser compartido, así como Twitter que tiene un teclado exclusivo para este tipo de contenido.
* Ayudan a construir una imagen moderna de tu negocio, donde a mayor alcance e interacción, la promoción se vuelve más interesante, atrayendo nuevos seguidores.
* Aumentan el tiempo de permanencia en tu sitio gracias al “autoloop”. Además, la reiteración del mensaje es muy efectiva para ayudar a que la información quede en la memoria.
* No sólo son fáciles de consumir, también son contenidos empáticos que ayudan a conectar con tus seguidores desde un enfoque emocional.
* **Desventajas:**
* No se recomienda para fotografías de alta calidad ni para imprimir, ya que solo tiene 256 colores.
* Su calidad es relativamente mala.
* Muy pocos colores y animaciones simples.
* Diseñarla toma más tiempo y recursos que una imagen fija en JPG, ya que se debe considerar la planificación y validación del relato junto a la ejecución de la imagen en distintos programas. De todas maneras, sigue siendo mucho más económico que un video, tanto en su ejecución como en la subida y carga.
* En campañas de mails, los problemas de compatibilidad de Outlook hacen que sólo aparezca el primer frame del GIF. Lo recomendado es crear los archivos de tal manera que muestren la información más relevante en este espacio.
* **PNG**

Es un formato de archivos de gráficos de mapa de bits (una trama). Fue desarrollado en 1995 como una alternativa gratuita al formato GIF, cuyos derechos pertenecen a Unisys (propietario del algoritmo de compresión LZW) y a quien los editores de software deben pagar regalías por usar este formato. Es muy común actualmente. Es ideal para gráficos web como iconos y funcionan muy bien con una paleta de color reducida.

* **Características:**
* El PNG utiliza un algoritmo de compresión sin perdida, por lo que reconstruye los datos de manera exacta a los originales. El problema con PNG es que, si se tiene una imagen con muchos colores y píxeles, obtendremos un archivo mucho más pesado que el JPG, por lo que no se usa con tanta frecuencia en Internet.
* El formato permite almacenar imágenes en blanco y negro (una profundidad de color de 16 bits por píxel) y en color real (una profundidad de color de 48 bits por píxel), así como también imágenes indexadas, utilizando una paleta de 256 colores.
* Soporta la transparencia de canal alfa, es decir, la posibilidad de definir 256 niveles de transparencia.
* Posee una función de entrelazado que permite mostrar la imagen de forma gradual.
* La compresión que ofrece este formato es (compresión sin pérdida) de 5 a 25 % mejor que la compresión GIF.
* Almacena información gama de la imagen, que posibilita una corrección de gama y permite que sea independiente del dispositivo de visualización. Los mecanismos de corrección de errores también están almacenados en el archivo para garantizar la integridad.
* **Ventajas:**
* Es excelente a la hora de editar y agregar efectos.
* Comprensión sin perdida.
* Permite generar imágenes de mapas de bits con una elevada 'ratio' de compresión de tamaño. Se trata de una compresión sin pérdida de calidad, al igual que el formato GIF, pero con mayor porcentaje de compresión media.
* Posee dos modos de compresión: estándar y progresivo. Este último facilita el reconocimiento de la imagen en los primeros instantes de la descarga de Internet, pues va mostrando líneas de la misma distribuidas por toda la superficie disponible. Se trata de una opción interesante cuando son imágenes muy grandes o se usan conexiones lentas.
* **Desventajas:**
* No sirve para la realización de animaciones complejas.
* Produce archivos más pesados que jpg.
* Al ser un formato sin pérdida de calidad, produce archivos excesivamente grandes para la reproducción de fotografías o cuadros.
* **SVG**

(Scalable Vector Graphics) Es un formato vectorial poco conocido pero muy útil para su uso online por su flexibilidad y por la capacidad de ofrecer gráficos con calidad. Es un formato gráfico basado en XML para crear archivos vectoriales en 2D, con un lenguaje de marcado por medio de etiquetas.

* **Características:**
* **Vectorial:** Es vectorial, lo que supone tener todas las ventajas de cualquier formato vectorial. Es escalable, pesa poco y permite una definición mayor a tamaños reducidos, mucho mayor que los archivos bitmap. El formato es igual al que se utiliza con cualquier programa vectorial como Corel Draw o Adobe Illustrator.
* **Standard abierto y compatible:** Es un formato abierto, estándar y basado en XML. Aunque las primeras versiones no se podían ver en los diferentes navegadores, hoy ya es un estándar que funciona sin problemas en todos los navegadores. SVG se convirtió en una recomendación del W3C en septiembre de 2001 con lo que en estos momentos ya es admitido por todos. Al ser un formato basado en XML necesitamos cierto control de código para hacer que un archivo SVG funcione adecuadamente.
* **Tipografías:** El formato SVG nos permite por un lado utilizar las tipografías con trazados, pero también nos permite incluirlas dentro del propio archivo en formato TrueType y Tipo 1, lo que nos da una capacidad extraordinaria en cuanto a que los motores de búsqueda son capaces de indexarlo. Hay que tener en cuenta que para el texto puro colocado como tipografía tiene que ser o bien de las fuentes instaladas en el sistema o bien incluidas como estilo CSS.
* Proyectos como Iconic también usan el formato SVG para hacer que una tipografía con iconos nos sirva para los diferentes usos y necesidades de tamaño y variación en función de las acciones que deseemos.
* **Interactivo:** Los archivos SVG pesan poco, igual que lo haría un archivo vectorial. Soportan estilos CSS, con lo que si cambiamos el estilo en nuestra web también cambiará dentro del archivo. Puede incluir scripts que permiten caminos dinámicos. Puede admitir acciones como los rollovers o cambios al hacer click.
* **Iconos responsive:** En la actualidad todas las webs están migrando al formato responsive, así que nos encontramos multitud de iconos que se redimensionan en función del ancho de la pantalla. Normalmente el funcionamiento suele ser sustituir un icono bitmap por otro más pequeño. Con SVG podemos utilizar el mismo y que se vaya reduciendo poco a poco, o bien que utilizar varios tamaños pero que todos ellos sean SVG con la ventaja de la mejor visualización al ser vectorial.
* Es fácil de usar, ya que se basa en el lenguaje XML, es decir el mismo lenguaje de etiquetas que usa HTML.
* Podemos por tanto crear las figuras y dibujos con SVG mediante etiquetas que incorporamos al lenguaje HTML.
* Incluir elementos SVG en la página se puede hacer desde el mismo código HTML. SVG dispone de una serie de etiquetas y atributos para introducir los elementos.
* Al estar el código SVG formado por los mismos elementos que el HTML podemos también aplicarle los lenguajes CSS y javascript.
* **Ventajas:**
* La gran ventaja de este formato de imagen es que se puede visualizar en navegadores. Esto es genial para las nuevas webs responsive.
* **Tamaño pequeño.** Los elementos de una imagen SVG abarcan mucho menos espacio que sus gemelos creados en formato ráster.
* **Flexibilidad.** Con CSS, puedes cambiar rápidamente la configuración de los gráficos en el sitio, tal y como el color de fondo o la posición del logotipo en la página. Para hacer esto, puedes editar el archivo en cualquier editor de texto
* Escalabilidad sin cambiar la calidad de la imagen.
* **Desventajas:**
* El gran problema que presenta es que está basada en el lenguaje de programación .xml, por lo que se necesita conocimientos sobre este lenguaje para hacerlo funcionar debidamente.
* El tamaño del archivo crece muy rápido si el objeto consiste de un gran número de pequeños elementos.
* Es imposible leer una parte del objeto gráfico, solamente se puede leer el objeto entero y eso baja tu velocidad.
* **Comparativa** (Santiago, s.f.)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Formato de imagen | Compresión | Paleta de colores | Transparencia | Animación | Uso |
| diferencias formatos imagen jpg | Con pérdidas | 16 millones de colores |  |  | Imágenes sin animación Fotografía |
| diferencias formatos imagen gif | Sin pérdidas | Máximo 256 colores | Binario |  | Animaciones simples Gráficos con colores planos Gráficos sin degradados |
| diferencias formatos imagen png 8 | Sin pérdidas | Máximo 256 colores |  |  | Uso similar al formato .GIF Mejor transparencia, pero sin animación Es perfecto para iconos |
| diferencias formatos imagen svg | Vector  Sin pérdidas | Millones de colores |  |  | Gráficos y logotipos para páginas web Pantallas de alta resolución |

## PROTOCOLOS DE SEGURIDAD

* **Secure Socket Layer (SSL)**

Secure Sockets Layer (SSL) es el estándar mundial de la seguridad en la Web. La tecnología SSL se enfrenta a potenciales problemas derivados de la visualización no autorizada de información confidencial, la manipulación de datos, la apropiación de datos, el phishing y los demás tipos de amenazas en los sitios Web. Para ello, se cifra la información confidencial a fin de que sólo los destinatarios autorizados puedan leerla. Además de evitar la manipulación de la información confidencial, SSL contribuye a que los usuarios tengan la seguridad de acceder a un sitio Web válido.

Los principales sistemas operativos, aplicaciones Web y hardware del servidor son compatibles con SSL, lo que significa que esta poderosa tecnología de cifrado de SSL ayuda a implementar en cada empresa una manta de seguridad que limita la responsabilidad para todo el sistema con el fin de afianzar la seguridad de los clientes, incrementar el porcentaje de transacciones finalizadas y optimizar los resultados finales. Gracias a los avances recientes obtenidos en la tecnología SSL, existe una amplia variedad de tipos de SSL. (Jromero, 2010)

* **Transport Layer Security (TLS)**

Para intentar corregir las deficiencias observadas en SSL v3 se buscó un nuevo protocolo que permitiera transacciones seguras por Internet, sobre todo teniendo en cuenta que SSL es propiedad de la empresa Nestcape. El resultado de esta búsqueda fue el protocolo TLS, que permite una compatibilidad total con SSL siendo un protocolo público, estandarizado por el IETF.

TLS busca integrar en un esquema tipo SSL al sistema operativo, a nivel de la capa TCP/IP, para que el efecto "tunel" que se implementó con SSL sea realmente transparente a las aplicaciones que se están ejecutando. (Jromero, 2010)

* **Protocolo S-HTTP**

El protocolo Secure HTTP fue desarrollado por Enterprise Integration Technologies, EIT, y al igual que SSL permite tanto el cifrado de documentos como la autenticación mediante firma y certificados digitales, de este modo, el usuario tiene ciertas garantías de que la información que envíe desde dicha página no podrá ser interceptada y utilizada por terceros. pero se diferencia de SSL en que se implementa a nivel de aplicación. Se puede identificar rápidamente a una página web servida con este protocolo porque la extensión de la misma pasa a ser .shtml en vez de .html como las páginas normales. (Jromero, 2010)

* **Protocolo SET**

SET se basa en el uso de certificados digitales para asegurar la perfecta identificación de todas aquellas partes que intervienen en una transacción on-line basada en el uso de tarjetas de pago, y en el uso de sistemas criptográficos de clave pública para proteger el envío de los datos sensibles en su viaje entre los diferentes servidores que participan en el proceso. Con ello se persigue mantener el carácter estrictamente confidencial de los datos, garantizar la integridad de los mismos y autenticar la legitimidad de las entidades o personas que participan en la transacción, creando así un protocolo estándar abierto para la industria que sirva de base a la expansión del comercio electrónico por Internet. (Jromero, 2010)

* **Protocolo SSH**

SSH (o Secure SHell) es un protocolo que facilita las comunicaciones seguras entre dos sistemas usando una arquitectura cliente/servidor y que permite a los usuarios conectarse a un host remotamente. A diferencia de otros protocolos de comunicación remota tales como FTP o Telnet, SSH encripta la sesión de conexión, haciendo imposible que alguien pueda obtener contraseñas no encriptadas.

SSH está diseñado para reemplazar los métodos más viejos y menos seguros para registrarse remotamente en otro sistema a través de la shell de comando, tales como telnet o RSH. Un programa relacionado, el SCP, reemplaza otros programas diseñados para copiar archivos entre hosts como RCP. Ya que estas aplicaciones antiguas no encriptan contraseñas entre el cliente y el servidor, evite usarlas mientras le sea posible. El uso de métodos seguros para registrarse remotamente a otros sistemas reduce los riesgos de seguridad tanto para el sistema cliente como para el sistema remoto.

## VULNERABILIDADES

Una vulnerabilidad es una debilidad del sistema informático que puede ser utilizada para causar un daño. Sin embargo, que exista una vulnerabilidad no significa que se produzca un daño en el equipo de forma automática. Es decir, la computadora tiene un punto débil, pero no por eso va a fallar, lo único que ocurre es que es posible que alguien ataque el equipo aprovechando ese punto débil.

Las vulnerabilidades de los sistemas informáticos las podemos agrupar en función de: (Gobierno de España. Ministerio de Educación, cultura y deporte, s.f.)

* **Diseño**
* Debilidad en el diseño de protocolos utilizados en las redes.
* Políticas de seguridad deficientes e inexistentes.
* **Implementación**
* Errores de programación.
* Existencia de “puertas traseras” en los sistemas informáticos.
* Descuido de los fabricantes.
* **Uso**
* Configuración inadecuada de los sistemas informáticos.
* Desconocimiento y falta de sensibilización de los usuarios y de los responsables de informática.
* Disponibilidad de herramientas que facilitan los ataques.
* Limitación gubernamental de tecnologías de seguridad.

### Tipos de vulnerabilidades

* **Autenticación rota**

Se presenta cuando es posible suplantar la identidad del usuario al obtener acceso a datos como contraseñas o identificadores. Un ejemplo es poder modificar el id de la sesión en la cookie y obtener así acceso como un administrador o cambiar el perfil de acceso.

* **Plan de contingencia:**
* Verificar los procesos de autentificación de usuario.
* Recomendar al usuario el cierre de sesión antes de cerrar el navegador por completo, implementar una ventana emergente con la siguiente frase: “¿Estás seguro de que deseas salir, de este sitio sin cerrar sesión?” De esa manera evitaremos que el usuario olvide cerrar su sesión por descuido.
* **Fallas al restringir acceso URL**

Una página en nuestro sitio que no sea no validada puede permitir el acceso áreas restringidas mediante la manipulación de la URL otorgando permisos administrativos a un atacante.

* **Plan de contingencia:**
* Asegurarnos de validar cada una de las paginas o controladores del sitio, utilizar métodos de autentificación incorporados en el servidor.
* **Configuración errónea de seguridad**

Un parámetro mal especificado puede permitir el acceso a nuestras aplicaciones.

* **Plan de contingencia:**
* Definir todos los elementos de seguridad y no usar atributos por defecto (por ejemplo, el usuario y password root), mantener nuestras aplicaciones, servidores y librerías siempre actualizados.
* **Inyección** **SQL**

Ocurre cuando a nuestro sistema entra información no confiable a través de formularios (comandos que son interpretados en nuestra base de datos). Puede resultar en robo o pérdida de nuestra información.

Va relacionado con los errores de formato de cadena (format string bugs) donde la principal causa de estos errores es aceptar sin validar la entrada de datos proporcionada por el usuario.

* **Plan de contingencia:**
* Validar y limpiar todo lo que el usuario ingrese a nuestro sistema antes de realizar cualquier proceso además de usar Prepared statements y stored procedures.
* **Almacenamiento inseguro**

Si un atacante tuviera acceso a nuestra información y esta no se encontrará asegurada, podría acceder a contraseñas y datos de tarjeta de crédito de usuarios y clientes entre otra información sensitiva.

Este tipo de problemas va relacionado también con la Insuficiente protección en la capa de transporte. Todo el tráfico en internet puede ser escuchado, y al enviar información sensitiva como contraseñas, números de tarjeta o documentos sin su apropiada autenticación y encriptación, alguien puede tener acceso a esa información.

* **Plan de contingencia:**
* Encriptar información sensible en nuestra base de datos.
* Usar SSL y TSL y evitar enviar información sensitiva que pueda ser accedida por un escuchante.

## ARRANQUE DEL PROYECTO

### Carta de inicio del proyecto

Es un documento emitido por el iniciador del proyecto o patrocinador, que autoriza formalmente la existencia de un proyecto y confiere al director del proyecto la autoridad para asignar los recursos de la organización a las actividades del proyecto. Documenta las necesidades de negocio, los supuestos, las restricciones, el conocimiento de las necesidades y requisitos de alto nivel del cliente y el nuevo producto, servicio o resultado que el proyecto debe proporcionar.

* **Ejemplo:**



### Perfiles de los participantes y del administrador

Para llevar a cabo un proyecto, sus ejecutores suelen tener diferentes responsabilidades a fin de facilitar la organización de las actividades encaminadas al cumplimiento de los objetivos. De igual manera, para el proceso de software las actividades son asignadas con base a la ingeniería del software, de acuerdo a las funciones requeridas en el modelo implementado, estos son los llamados roles en la ingeniería de software. (RECIBE, 2013)

Definir los roles a asumir por los integrantes de un equipo de proyecto, teniendo en cuenta las características de la industria de software nacional, constituye un punto de partida válido para contribuir a su desarrollo.

La efectividad de los equipos de trabajo tiene una gran importancia para el funcionamiento de la empresa. En la actualidad una buena gestión del equipo directivo puede suponer una ventaja competitiva frente a las empresas competidoras. Y en el producto final se ve reflejado un software de calidad.

* **Ejemplo:**



### Matriz de responsabilidad

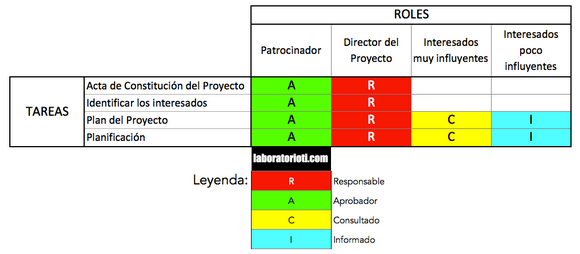
La matriz de asignación de responsabilidades es un mapa estructural que sirve para establecer las responsabilidades de cada actor que participa en una tarea.

Conocer los roles y responsabilidades de cada uno de los integrantes de una organización es esencial para ejecutar los procesos de trabajo con agilidad y eficiencia además de evitar clientes insatisfechos, oportunidades de negocio perdidas, confusión y mal clima interno, pérdidas de tiempo, trabajadores insatisfechos e improductivos, etc.

La importancia de la matriz recae en que ayuda a establecer expectativas, y a asegurar que las personas saben y entienden lo qué se espera de ellas.

La metodología RACI es una de las más comunes al momento de realizar una matriz de responsabilidades.

* **El significado de las siglas RACI**
* **“R” – Responsable** (del inglés Responsabile). Es el rol encargado de realizar la tarea. Hay que tener cuidado que en este caso responsable significa responsable de realizar el trabajo no que sea el que tiene capacidad de decidir cuando está completo o no.
* **“A” – Aprobador** (del inglés Accountable y algunas veces Approver). Es el rol que aprueba el trabajo realizado por el responsable.
* **“C” – Consultado** (del inglés Consulted). Son aquellas personas que son consultadas sobre la cuestión, personas a las cuales se les pregunta su opinión sobre algún aspecto de la tarea ya bien sea porque deben tenerse en cuenta o porque son expertos en la materia.
* **“I” – Informado** (del inglés Informed). Son aquellas personas a las que hay que mantener informadas sobre la evolución de la tarea. Lo más frecuente es informar de cuando se ha completado la tarea, pero dependiendo del rol y de la implicación pueden solicitar que se les informe de la evolución.
* **Reglas a tener en cuenta en las matrices RACI**
* Al menos debe haber un responsable dentro de una matriz RACI.
* Sólo puede haber un Aprobador por tarea dentro de la matriz RACI.
* Puedes incluir tantos responsables, Informados y Consultados como necesites.
* **Pasos para aplicar la matriz RACI:**
* Identificar las actividades de algún proceso (y colocarlas como filas de la matriz).
* Identificar / definir los principales roles funcionales (y colocarlos como columnas de la matriz).
* Asignar los códigos “RACI” a cada tarea.
* Identificar ambigüedades o problemas (solapamientos, vacíos, dudas, etc.) y trabajar para solucionarlos.
* Distribuir la matriz e incorporar el feedback.
* Comunicarla de modo efectivo a todos los involucrados en el proceso.
* Una matriz está viva y hay que asegurar que se haga una actualización periódica.
* Un mismo rol puede ser compartido por más de una persona o viceversa, sobre todo en organizaciones pequeñas.
* **Ejemplo:**



### Lista de factores de desempeño

* **Definición:**

La evaluación del desempeño es un proceso sistemático y periódico de estimación cuantitativa y cualitativa del grado de eficacia con el que las personas llevan a cabo las actividades y responsabilidades de los puestos que desarrollan.

Su finalidad es juzgar o estimar el valor, la excelencia y las cualidades de una persona y, sobre todo, su contribución a la organización. Así se consigue determinar problemas de supervisión y gerencia, integración de personas a la organización, adecuación de la persona al cargo, posibles disonancias o falta de entrenamiento. Para, en consecuencia, poder establecer los medios y programas para eliminar o neutralizar tales problemas, mejorando la calidad del trabajo y la calidad de vida en las organizaciones.

La organización internacional ISO, ha publicado en 2012 la norma ISO 10667: 2012. Procedimientos y métodos para la evaluación de personas en entornos laborales y contextos organizacionales. Esta norma tiene como misión proporcionar una guía clara y concisa para unas buenas prácticas en evaluación de personas en contextos laborales, y lo hace desde una perspectiva basada en evidencias, medible y aplicable a nivel mundial. Se estructura en dos partes:

Un primer referente a los requisitos para el cliente y la segunda, a los requisitos para los proveedores del servicio de evaluación.

* **Importancia:**

Las principales razones para evaluar el desempeño de los empleados en una organización son las siguientes:

* Proporciona un juicio sistemático para fundamentar aumentos salariales, promociones y, en otras ocasiones, despidos.
* Permite comunicar a los empleados cómo marchan en su trabajo y qué deben cambiar en su comportamiento, habilidades o conocimientos.

La norma describe las competencias, obligaciones y responsabilidades de los clientes y de los proveedores, antes, durante y después del proceso de evaluación, y proporciona directrices para todas las partes implicadas en el proceso evaluador y cubre todo el ciclo de la vida laboral, a escala individual, grupal y organizacional: desde la selección hasta el desarrollo profesional, pasando por la formación, el clima laboral, etc.

* **Ejemplo:**



### Esquema del ciclo de vida

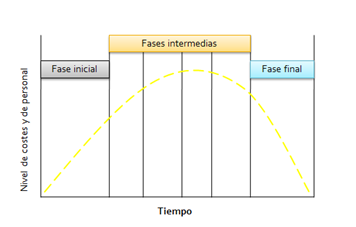
* **Definición:**

El ciclo de vida del proyecto define las fases que conectan el inicio de un proyecto con su fin. Un ciclo de vida para un proyecto se compone de fases sucesivas compuestas por tareas panificables.

La transición de una fase a otra dentro del ciclo de vida de un proyecto generalmente implica y, por lo general, está definida por alguna forma de transferencia técnica. Generalmente, los productos entregables de una fase se revisan para verificar si están completos, si son exactos y se aprueban antes de iniciar el trabajo de la siguiente fase.

No obstante, no es inusual que una fase comience antes de la aprobación de los productos entregables de la fase previa, cuando los riesgos involucrados se consideran aceptables.(UCAB, 2016)

* **Fases de un proyecto:**
* **Fase Inicial**
* **Fase conceptual**: Es la etapa donde nace la idea, se formula el proyecto al analizar los puntos clave, se toma la decisión favorable de iniciar actividades del proyecto, se establecen las metas, se hacen los principales nombramientos y asignaciones de recursos.
* **Fases Intermedias**
* **Fase organizacional**: Contempla el período de planificar e idear la mejor forma de hacer realidad lo planteado en la fase conceptual. Se diseña la organización y constituye el equipo de proyecto, se buscan los recursos y se hace el plan maestro y detallado de actividades.
* **Fase ejecutiva:** En esta etapa es donde se ejecutan los trabajos principales del proyecto como el desarrollo de los programas, la construcción de las instalaciones, las pruebas, las entregas, etc.
* **Fase Final**
* **Fase de completación:** Es el período donde se terminan las actividades, se cierran los contratos se transfieren los recursos y compromisos a otras organizaciones, se hace la puesta en marcha, etc.



* **Importancia:**

El ciclo de vida del proyecto es de mucha importancia porque define las fases que conectan el inicio de un proyecto con su fin. Un ciclo de vida para un proyecto se compone de fases sucesivas compuestas por tareas panificables.

La transición de una fase a otra dentro del ciclo de vida de un proyecto generalmente implica y, por lo general, está definida por alguna forma de transferencia técnica.

Generalmente, los productos entregables de una fase se revisan para verificar si están completos, si son exactos y se aprueban antes de iniciar el trabajo de la siguiente fase. No obstante, no es inusual que una fase comience antes de la aprobación de los productos entregables de la fase previa, cuando los riesgos involucrados se consideran aceptables.

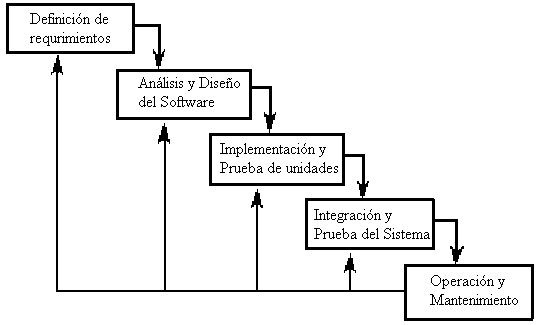
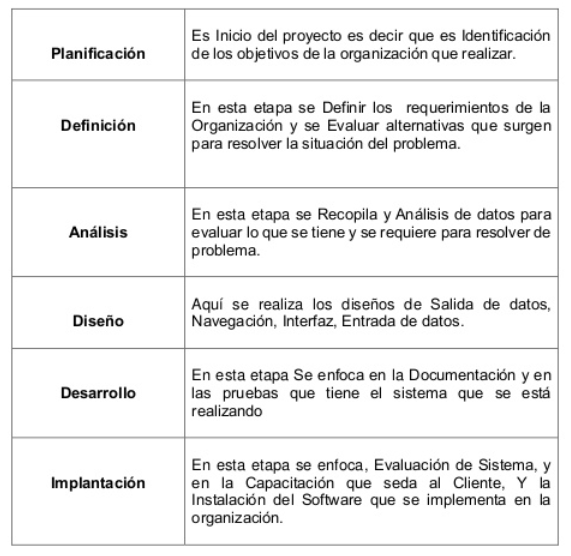


Figura 5 Modelo de cascada retroalimentado

### Justificación de las actividades del proceso

Se utiliza la justificación de las actividades para tener un orden adecuado y fácil de entender para elabora el proyecto más fácil siguiendo una serie de pasos definidos, además de tener en claro que se realiza en cada una de las actividades del ciclo de vida.

* **Ejemplo:**



### Diagrama de Gantt

Es una herramienta visual para la **planificación** y programación de actividades o tareas sobre una línea del tiempo. Permite al usuario establecer la duración y el comienzo de cada actividad. A través de una gráfica, fácil de interpretar, el usuario puede llevar un control de la planificación de su trabajo.

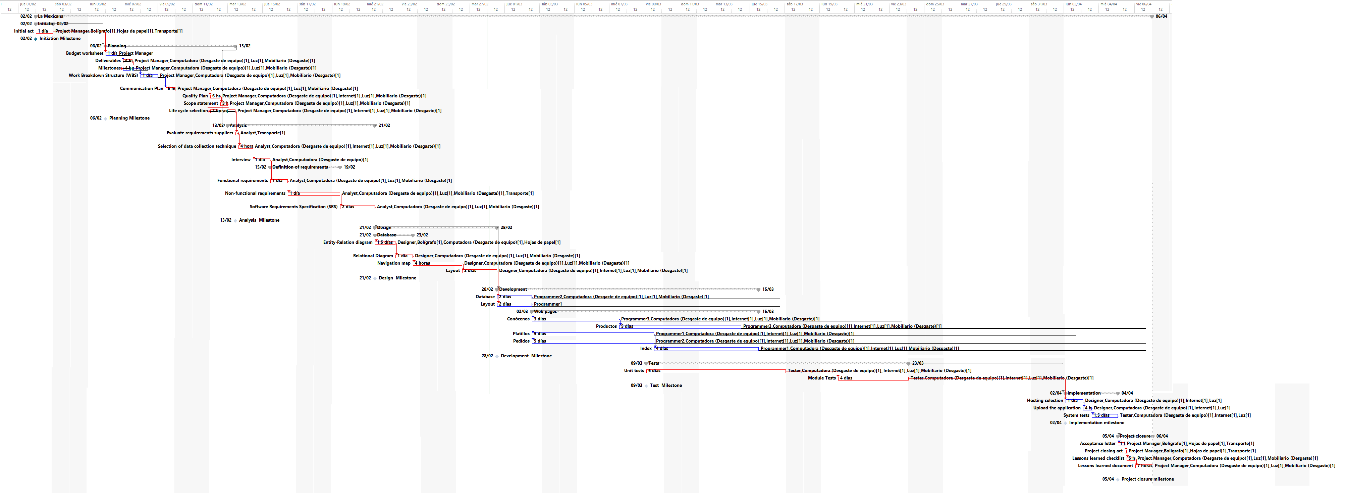
* **Importancia:**

Se obtiene unaimagen relativamente simple de un sistema complejo. Es decir que, de forma muy visual, se nos pone delante una gráfica que refleja la organización de las fases de un proyecto. Aquí la frase “una imagen vale más que mil palabras” es del todo aplicable, ya que la visualización del diagrama de Gantt facilita enormemente la comprensión de todo el proceso.

Ayuda a organizar las ideas. Cuando los objetivos y las acciones se dividen en segmentos más pequeños resultan más accesibles, más fáciles de alcanzar. A la vez, se ve más clara su posible complejidad. La construcción de un diagrama de Gantt obliga a seccionar el todo en diferentes partes y crea un cuadro con todas sus piezas.

Contribuye a establecer plazos realistas. Las barras del gráfico indican en qué período se completará una tarea o un conjunto de tareas. Permite tomar una perspectiva temporal adecuada y es útil para la consecución a tiempo de los objetivos fijados. Importante es también tener en cuenta otros eventos de la compañía ajenos al proyecto, que podrían consumir también recursos y tiempo.

Resulta de gran utilidad para otros departamentos no involucrados en el proyecto. Como el diagrama de Gantt es una herramienta muy gráfica, cualquier persona puede comprender fácilmente cuáles son las etapas del proceso. Situarlo en un lugar visible y en formato grande, donde todo el mundo pueda verlo, ayudará a que se recuerden los objetivos y se conozca cuándo van a tener lugar las acciones planificadas.



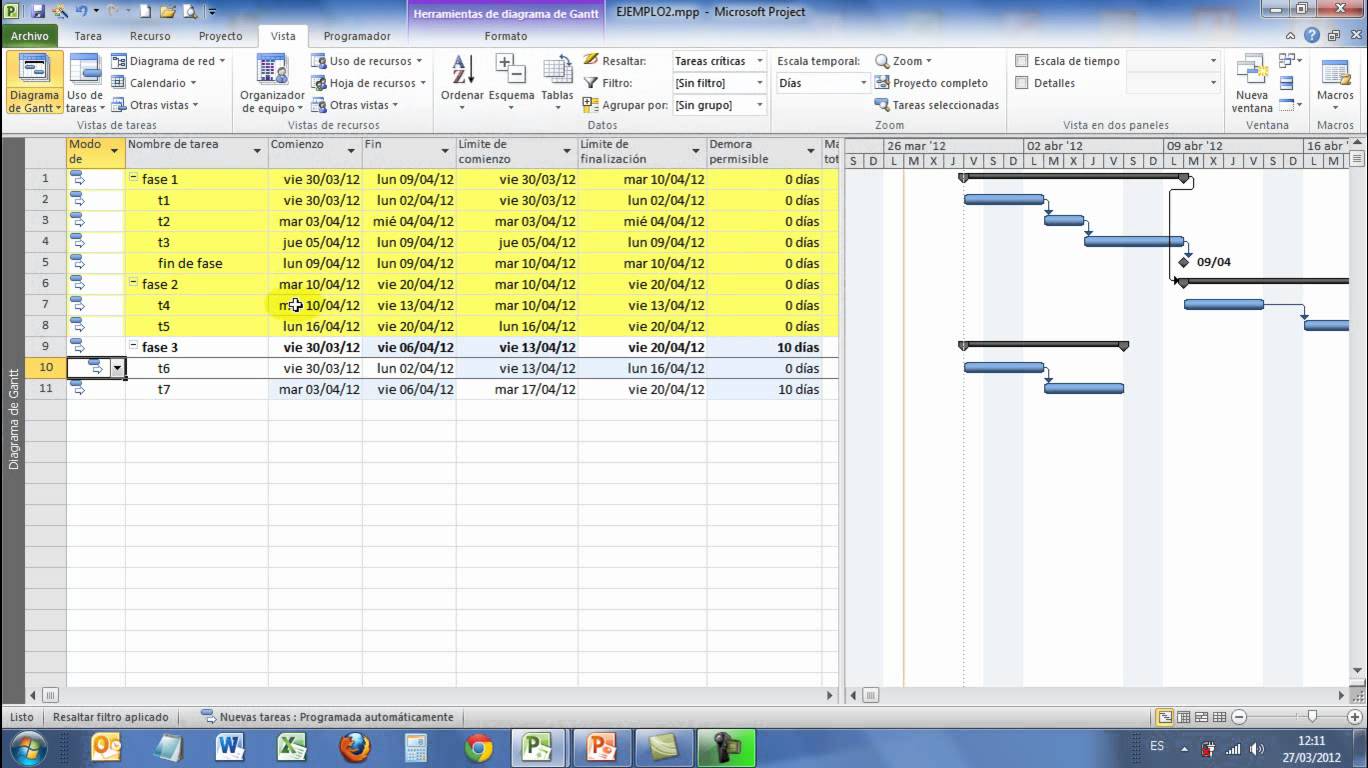
### Diagrama de ruta crítica

El **método de la ruta crítica CPM (Critical Path Method),** es un algoritmo basado en la teoría de redes diseñado para facilitar la planificación de proyectos, en el que ingresa todos los factores necesarios que intervienen en su proyecto y genera la línea de tiempo óptima para completarlo.

Con ella, **sabremos la duración total del mismo y el estado de urgencia de las actividades** marcadas en un cronograma.

* **Importancia:**

Nos permite definir el tiempo más temprano que se pueden entregar los proyectos (auto-realización eficaz), definir la serie de actividades que tienen menor flexibilidad en un proyecto (actividades que no debemos posponer sino en estrictas circunstancias), conocer las holguras del proyecto (actividades que podemos posponer por algunos cafés sin que afecten el cumplimiento – en tiempo – de nuestro objetivo o proyecto).



# REFLEXIONES

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Integrantes** |  |  |  |  | **Total** |
| 1. Jonathan Jair Alfaro Sánchez | 9 | 9 | 9 | 9 | 9 |
| 2. Mary Carmen Crescencio Bernal | 9 | 9 | 9 | 9 | 9 |
| 3. Miguel Ángel Mandujano Barragán | 9 | 9 | 9 | 9 | 9 |
| 4. Giovanni Misael Alfaro Sánchez | 9 | 9 | 9 | 9 | 9 |

**Jonathan Jair Alfaro Sánchez**

* ¿**Por qué me auto-evaluó de esta forma?**

Considero que en esta actividad el trabajo fue muy equitativo, por lo cual tanto yo como mis compañeros nos autoevaluamos con un 9.

* **¿Por qué co-evaluó a mis compañeros de esta forma?**

Todos trabajamos bien, entregando todo a tiempo y en forma, por eso considero que todos debemos obtener la misma calificación.

**Mary Carmen Crescencio Bernal**

* **Por qué me auto-evaluó de esta forma?**

Considero que, en esta actividad mi desempeño fue bueno porque participe, realice y entregue las tareas que me fueron asignadas en tiempo y en forma para evitar retrasos.

* **¿Por qué co-evaluó a mis compañeros de esta forma?**

Considero que el reparto de tareas en esta actividad fue muy equitativo por tal razón, pienso que cada uno de mis compañeros trabajo de la misma manera, y todos cumplieron con cada una de las actividades que les fueron asignadas.

**Miguel Ángel Mandujano Barragán**

* **Por qué me auto-evaluó de esta forma?**

Considero que aporte la información que me correspondía muy completa y verídica, usando diferentes fuentes confiables de internet.

* **¿Por qué co-evaluó a mis compañeros de esta forma?**

Considero que mis compañeros trabajaron de manera excelente en esta actividad, aportando información importante en todo momento para finalmente terminar de manera exitosa la actividad.

**Giovanni Misael Alfaro Sánchez**

* **Por qué me auto-evaluó de esta forma?**

Creo que como miembro del equipo aporte considerablemente a la actividad, además de que investigue demasiadas cosas dentro de mis tareas asignadas y pude realizar un buen trabajo.

* **¿Por qué co-evaluó a mis compañeros de esta forma?**

Considero que su esfuerzo para esta actividad fue sobresaliente y se merecen una buena calificación, también creo que su manera de trabajar es muy dedicada y ponen mucho de su parte para entregar una buena actividad.

# PROBLEMAS PRESENTADOS

Durante la realización de esta actividad se presentaron varios problemas, entre ellos la falta de información de algunos temas que se tenían que cubrir, en la sección de “Arranque de proyecto” se nos pidió poner definiciones e importancia de algunos documentos y anexar un ejemplo o plantilla de cualquier otro proyecto. Desafortunadamente no encontramos ejemplos de algunos documentos, por lo cual Mary propuso realizar una investigación para poder realizar nuestro propio ejemplo.

Otro de los problemas externos que se nos presentaron fue para la realización de las tecnologías que fueron expuestas en clase porque, algunos de nuestros compañeros tardaron en compartir su información, por lo cual Jonathan y Misael propusieron investigar los conceptos que nos hacían falta.

Encontramos información muy ambigua ya que para la sección de medios digitales nos enfocamos a buscar solo formatos que sean compatibles con aplicaciones web, pero fue difícil encontrar solo algunos de los formatos más recomendados para web sobre todo para los archivos de video, Miguel propuso seguir buscando información en sitios diferentes y comparar la información.

# BIBLIOGRAFÍA

Albéniz, I. M. (23 de 04 de 2015). *Linkedin*. Recuperado el 19 de 05 de 2018, de Matriz RACI : definiendo roles y responsabilidades: https://es.linkedin.com/pulse/matriz-raci-definiendo-roles-y-responsabilidades-i%C3%B1aki-albeniz

DYMS, P. m. (18 de Agosto de 2016). *MDN web docs moz://a*. Obtenido de CSS3: https://developer.mozilla.org/es/docs/Web/CSS/CSS3

*EcuRed*. (s.f.). Recuperado el 15 de 05 de 2018, de Imagen digital: https://www.ecured.cu/Imagen\_digital

Gallardo López Domingo, P. P. (31 de Enero de 2008). *Biblioteca Digital ECEST.* Obtenido de Rua - Lenguajes de programación: http://hdl.handle.net/10045/4030

*Gobierno de España. Ministerio de Educación, cultura y deporte*. (s.f.). Recuperado el 17 de 05 de 2018, de Seguridad infromática. Vulnerabilidades de un sistema informático: http://descargas.pntic.mec.es/mentor/visitas/demoSeguridadInformatica/vulnerabilidades\_de\_un\_sistema\_informtico.html

Gómez, J. (22 de 02 de 2016). *TICcionario*. Recuperado el 19 de 05 de 2018, de ¿Qué es una matriz RACI y cómo usarla?: http://www.laboratorioti.com/2016/02/22/ticcionario-una-matriz-raci-usarla/

*HTML5*. (s.f.). Obtenido de Capítulo 1 ¿Qué es HTML5?: https://www.arkaitzgarro.com/html5/capitulo-1.html

JorgeAML, e.-c.-d. r. (17 de Enero de 2018). *MDN web docs moz://a*. Obtenido de ¿Qué es JavaScript?: https://developer.mozilla.org/es/docs/Learn/JavaScript/First\_steps/Qu%C3%A9\_es\_JavaScript

Jromero. (13 de Enero de 2010). *todotecnology.blogspot.* Obtenido de todotecnology.blogspot - Protocolos de seguridad Web: http://todotecnology.blogspot.mx/2010/01/protocolos-de-seguridad-web.html

Noriega, S. (14 de Noviembre de 2015). *CertSuperior.* Obtenido de CertSuperior - Protocolos de Seguridad Básicos ¿Qué tan seguro es tu sitio web?: https://www.certsuperior.com/Blog/protocolos-de-seguridad-sitio-web-seguro

Padrón, O. F. (2008). *Biblioteca Ecest.* Obtenido de Sistema informático para gestionar Expedientes de Proyectos (SIGEP): https://doaj.org/article/7b5f26a2675b42c8be96bf2f49ace89f

*PowerData Especialistas en Gestión de Datos*. (26 de 05 de 2016). Recuperado el 15 de 05 de 2018, de ¿Qué es un gestor de datos y para qué sirve?: https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/que-es-un-gestor-de-datos-y-para-que-sirve

Puertas, J. P. (2004). *Creación de un portal con PHP y MYSQL.* México: RA-MA Editorial.

*RECIBE*. (04 de 04 de 2013). Recuperado el 18 de 05 de 2018, de Influencia de los Roles de Equipo en las Actividades del Desarrollador de Software: http://recibe.cucei.udg.mx/revista/es/vol2-no1/computacion02.html

Santiago, I. (s.f.). *Ignacio Santiago*. Recuperado el 15 de 05 de 2018, de GIF vs PNG vs JPG vs SVG ¿Qué FORMATO de IMAGEN es Mejor?: https://ignaciosantiago.com/comparativa-formatos-imagenes-png-jpg-gif-svg/

UCAB. (2016). *Ciclo de vida del proyecto de TI* . Obtenido de https://sites.google.com/site/gpsguayana/ciclo-de-vida-del-proyecto-de-ti

*Univia blogs*. (13 de 03 de 2015). Recuperado el 15 de 05 de 2018, de Base de Datos 1. Oracle vs MySQL vs SQL Server: una comparación entre los Sistemas Gestores de Bases de Datos Relacionales más Populares: https://basededatosunounivia.wordpress.com/2015/03/13/oracle-vs-mysql-vs-sql-server-una-comparacion-entre-los-sistemas-gestores-de-bases-de-datos-relacionales-mas-populares/

# DIAGRAMA DE ACTIVIDADES

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Actividad** | **Nombre** | **Fechas propuestas** | | **Fechas reales** | | **Responsable** | **Dificultades** | **Solución** | **Áreas de mejora** |
| **Inicio** | **Fin** | **Inicio** | **Fin** |
| 1 | Introducción | 17/05/2018 | 18/05/2018 | 19/05/2018 | 19/05/2018 | JJAS | Ninguna |  | La redacción. |
| 2 | Tecnologías web | 12/05/2018 | 15/05/2018 | 15/05/2018 | 17/05/2018 | MCCB, MAMB, JJAS, GMAS | Ninguna | Ninguna | Tomar más ideas de autores reconocidos. |
| 2.1 | Servidores | 12/05/2018 | 15/05/2018 | 12/05/2018 | 14/05/2018 | MCCB | Encontré muy poca información sobre el servidor de TOMCAT. | Buscar en diferentes fuentes de información. | Leer más la información. |
| 2.2 | Programación | 12/05/2018 | 15/05/2018 | 14/05/2018 | 15/05/2018 | MAMB | Ninguna | Ninguna | Leer más acerca de los lenguajes de programación de lado servidor. |
| 2.3 | Gestores de Bases de Datos | 12/05/2018 | 15/05/2018 | 15/05/2018 | 16/05/2018 | JJAS | Ninguna | Ninguna | Buscar más información acerca del tema para completar la investigación |
| 2.4 | Maquetado y diseño | 12/05/2018 | 15/05/2018 | 14/05/2018 | 16/05/2018 | GMAS | Ninguna | Ninguna | Buscar más información acerca del tema para completar la investigación |
| 2.5 | Nuevas tecnologías | 12/05/2018 | 15/05/2018 | 13/05/2018 | 14/05/2018 | MCCB | Algunos de los compañeros tardaron un poco en compartir su información. | Buscar la información. | Pedirles a los compañeros que compartan su información a tiempo. |
| 3 | Medios digitales |  |  |  |  |  |  |  |  |
| 3.1 | Audio | 12/05/2018 | 15/05/2018 | 14/05/2018 | 16/05/2018 | GMAS | Dificultad para encontrar información sobre audio en la web. | Realizar una lectura avanzada para obtener la mejor información sobre el audio en la web. | Realizar más investigaciones y hacer pruebas para verificar las ventajas y desventajas del audio en la web. |
| 3.2 | Video | 12/05/2018 | 15/05/2018 | 15/05/2018 | 16/05/2018 | MAMB | Difícil encontrar información relacionada a web | Buscar y leer en foros y sitios seguros. | Realizar una búsqueda más avanzada sobre video en la web. |
| 3.3 | Imagen | 12/05/2018 | 15/05/2018 | 15/05/2018 | 16/05/2018 | JJAS | Ninguna | Ninguna | Realizar pruebas para conocer las ventajas de integrar diferentes formatos de imagen a la web. |
| 4 | Protocolos de seguridad | 15/05/2018 | 18/05/2018 | 15/05/2018 | 16/05/2018 | MAMB, GMAS | Dificultad para encontrar información sobre que método utilizan los protocolos. | Se realizó una búsqueda muy completa de los protocolos. | Hacer más búsquedas en libros físicos. |
| 5 | Vulnerabilidades | 15/05/2018 | 18/05/2018 | 13/05/2018 | 17/05/2018 | MCCB, JJAS | Hay muy poca información acerca de este tema en internet. | Buscar información en diferentes sitios. | Realizar más búsquedas en libros físicos. |
| 6 | Arranque del proyecto |  |  |  |  |  |  |  |  |
| 6.1 | Carta de inicio del proyecto | 15/05/2018 | 18/05/2018 | 16/05/2018 | 19/05/2018 | MAMB | Ninguna | Ninguna | Comparar diferentes formatos para hacerlo de una forma más completa. |
| 6.2 | Perfiles de los participantes y del administrador | 15/05/2018 | 18/05/2018 | 18/05/2018 | 19/05/2018 | JJAS | Ninguna | Ninguna | Buscar información diferente para seleccionar la más adecuada. |
| 6.3 | Matriz de responsabilidad | 15/05/2018 | 18/05/2018 | 18/05/2018 | 19/05/2018 | JJAS | Confusión ya que es parecida a la matriz de comunicación. | Realizar una investigación para hacer la matriz de responsabilidades correctamente. | Buscar diferentes ejemplos para tener una matriz de responsabilidad más completa. |
| 6.4 | Lista de factores de desempeño | 15/05/2018 | 18/05/2018 | 15/05/2018 | 17/05/2018 | MCCB | No encontrar información suficiente ni ejemplos de este documento. | Buscar más información y realizar nuestra propia plantilla. | Buscar más información en libros físicos. |
| 6.5 | Esquema del ciclo de vida | 15/05/2018 | 18/05/2018 | 15/05/2018 | 17/05/2018 | MCCB | Ninguno. | Ninguno. | Buscar más información en libros físicos. |
| 6.6 | Justificación de las actividades del proceso | 15/05/2018 | 18/05/2018 | 16/05/2018 | 19/05/2018 | MAMB | No encontrar información asertiva sobre este apartado. | Realizar una búsqueda a fondo para encontrar mejores resultados. | Buscar una fuente confiable en internet a través de una búsqueda a fondo. |
| 6.7 | Diagrama de Gantt | 15/05/2018 | 18/05/2018 | 17/05/2018 | 19/05/2018 | GMAS | Ninguna | Ninguna | Realizar el diagrama de Gantt más detallado. |
| 6.8 | Diagrama de ruta crítica. | 15/05/2018 | 18/05/2018 | 17/05/2018 | 19/05/2018 | GMAS | Ninguna | Ninguna | Leer y practicar con el fin de identificar mejor la ruta crítica en un proyecto. |
| 7 | Reflexiones | 18/05/2018 | 19/05/2018 | 18/05/2018 | 19/05/2018 | JJAS, MCCB, MAMBM GMAS | Ninguna | Ninguna | Practicar mejor la redacción de nuestras ideas |