

Instituto Tecnológico de Tuxtla Gutiérrez Chiapas



CARRERA:

Lic. En Informática

NOMBRE DEL PROYECTO:

Desarrollo de una aplicación Web para consulta de Resultados clínicos y servicios de la Unidad Médica Ángel.

MATERIA:

Residencia Profesional

ALUMNO:

Leep Chanona Marco Antonio de Jesús

Semestre:

9

INDICE

Introducción	4
Justificación	5
Objetivo general	7
Objetivo específico	7
Caracterización del área en que participo	8
problemas a resolver	10
Cronograma de actividades	11
Descripción detallada de las actividades	11
Alcances y limitaciones	12
Fundamento teórico	13
1.1.-Antecedentes	
1.2.-Interfaz	
1.3.-Consideraciones técnicas	
1.4.-Estructura de las aplicaciones web	
1.5.-Ventajas	
1.6.-Inconvenientes	
1.7.-Diferencias entre aplicación web y aplicación de internet enriquecida (ria)	
1.8 Base de datos	
Aplicaciones	13
Antecedentes	13
Interfaz	14
Consideraciones técnicas	14
estructura de las aplicaciones web	15
Ventajas	16
Inconvenientes	17

diferencia entre aplicación web y aplicación de internet enriquecida	17
Procedimientos y descripción de actividades realizadas	18
Perl	19
Ruby	20
Python	20
Visual basic	20
HTML	21
Elementos HTML	22
Atributos HTML	23
XML	24
Base de datos	24
Base de datos estáticas	25
Base de datos dinámicas	25
Base de datos bibliográficas	26
Base de datos de texto completo	26
Directorios	26
Modelo de base de datos	27
Base de datos jerárquicas	28
Base de datos de red	28
Bases de datos transaccionales	28
Base de datos relacionales	29
Bases de datos multidimensionales	30
Bases de datos orientada a objetos	30
Conclusiones	32

Recomendaciones33

Bibliografía34

INTRODUCCIÓN

En los últimos años, por distintas causas, ha aumentado significativamente el índice de enfermedades, y por ende, la demanda de servicios en los centros de salud, pero debido a la obsolescencia en la administración de los mismos, el servicio proporcionado a los ciudadanos resulta ser de muy baja calidad, y con esto nos referimos, entre otras cosas, a la falta de disponibilidad de atención a los pacientes, el tardo diagnóstico de los resultados de análisis, la falta de disposición para resolver las dudas de los mismos, etc.

La Unidad Médica Ángel de Tuxtla Gutiérrez ubicado en Calle central entre 10 y 11 sur (frente a la facultad de medicina humana), esta unidad médica tiene un gran número de pacientes, actualmente cuenta con más de 30 consultas médicas por día. Al finalizar cada consulta se le pide al paciente volver en determinado tiempo por sus resultados mientras estos son procesados, sin embargo algunas veces los pacientes no regresan por los resultados de los análisis por falta de tiempo o por ciertos imprevistos lo que retarda su atención curación y satisfacción en tanto los pacientes que si acuden son atendidos y recetados inmediatamente.

Las herramientas tecnológicas se han convertido hoy en día en algo primordial para la resolución de problemas y ayudar a los pacientes brindándoles un mejor servicio y atención, se pueden establecer procesos de la relación entre usuario y la maquina con distintos objetivos (resultados clínicos y solicitud de citas), entre otras cosas como medio publicitario y así darse a conocer en sus servicios.

Los avances en la ciencia y la tecnología cambian el estilo de vida del ser humano debido a que actualmente la mayoría de las actividades del hombre giran en torno al uso de la Electrónica y Computación, los cuales han utilizado para crear, informar, brindar servicios etc. Hoy la tecnología está evolucionando muy rápido, lo que ha traído lo que se pensaba como imaginable a nuestro alcance.

México ha luchado por tener un nivel cercano al de los países desarrollados, pero en materia de salud, está muy por debajo de la calidad de los servicios de estos países, y el principal factor que propicia este problema, es la falta en el uso de la tecnología, para una mejor y pronta atención a los pacientes, por lo tanto, es necesaria la implementación de esta en todos los centros de salud, ya que con esto, no solamente mejoraría el servicio, sino la calidad de vida de todos los ciudadanos que lo requieran, trayendo como consecuencia, un mejor desempeño de los mismos en las distintas áreas donde se desarrollen.

Tomando en cuenta que hoy en día la población vive de prisa derivado de muchos factores sociales o económicos, la tecnología juega un papel muy importante, ya que su principal característica es minimizar procesos que ahorren tiempo al que la utiliza y muchas personas tienen acceso a la tecnología desde un teléfono celular y que mejor que mediante el uso de este puedan atender su salud para mejorar su calidad de vida. Sin descuidar su tiempo.

JUSTIFICACIÓN

Actualmente existen muchas áreas de consulta médica se abren, nuevos hospitales, laboratorios y farmacias donde brinden atención al público como consulta externa, sin embargo la publicidad y facilidades que se le brinden a los pacientes hacen la diferencia y logran que una empresa se destaque de los demás gracias a la tecnología.

Por esta razón se desarrollara una aplicación web para La Unidad Médica Ángel que ayudará tanto al médico como al paciente a tener un control de información, esto facilitará el manejo de información de los resultados clínicos y citas médicas de cada paciente y al mismo tiempo por medio de la aplicación, los pacientes podrán tanto solicitar una consulta médica como verificar e imprimir sus resultados de laboratorio

Con esto el paciente no tiene la necesidad de acudir nuevamente a recoger sus resultados luego después acudir al médico que lo recete lo que le ocasiona pérdida de tiempo y en algunas veces hace que posponga el cuidado de su salud.

Mediante la creación de la aplicación web el paciente solo debe acudir una sola vez a la unidad médica ángel y posteriormente consultar sus resultados en la página y si así lo desea incluso consultar la receta médica emitida por su médico tratante

Con la creación de la aplicación web se beneficia por una parte población abierta, tecnológicamente activa, empleados de diversos sectores tanto de gobierno como privado y al mismo tiempo la Unidad Médica Ángel que será la receptora de los pacientes.

OBJETIVO GENERAL

Desarrollar una aplicación Web que permita consultar los servicios de la Unidad Médica Ángel, así como resultados clínicos y solicitud de citas.

OBJETIVOS ESPECÍFICOS

- Mejorar el proceso de comunicación entre el paciente y el medico a través de la aplicación web.
- Dar a conocer a los pacientes y público en general todos los servicios que presta la unidad médica así como lo más relevante en cuanto a noticias, avisos médicos y otro tipo de información importante.
- Proporcionar una herramienta tecnológica que destaque y publiciteLa Unidad Médica Ángel.
- Facilitar el manejo de información de cada uno de los pacientes de La Unidad Médica Ángel.
- Permitir al paciente consultar sus resultados de laboratorio, consulta o cualquier servicio solicitado
- Proporcionar al paciente acceso a su historial médico mediante un acceso privado

CARACTERIZACIÓN EN EL ÁREA EN QUE PARTICIPO

La Unidad Médica Ángel de Tuxtla Gutiérrez se encuentra ubicado en calle central entre 10 sur y 11 sur (frente a la facultad de Medicina), es una empresa integrada por un equipo de médicos responsables y comprometidos a conservar y/o restaurar la SALUD integral del paciente dicha Unidad Médica cuenta con un cierto número de pacientes, el área de recepción es el área encargada de administrar información a los pacientes y no pacientes así como también hacer entrega de los resultados clínicos.



La Infraestructura Informática Consta De 3 Computadoras en el área administrativa y una computadora en el área de recepción todas con las mismas características:

Monitores LCD de 15 pulgadas marca Samsung

Cpu Intel core 5

Disco duro de 80 gigas

Memoria Ram de 1 gb

Teclado y mouse

PROBLEMAS A RESOLVER

Seguimiento al control de un paciente:

Al finalizar cada consulta se le pide al paciente volver en determinado tiempo por sus resultados mientras estos son procesados sin embargo algunas veces los pacientes no regresan por los resultados de los análisis por falta de tiempo o por ciertos imprevistos lo que retarda su atención curación y satisfacción

Conclusión para el paciente:

No todos los pacientes concluyen satisfactoriamente algunos porque no pudieron acudir por sus resultados y otros porque no acudieron al médico con sus resultados

Satisfacción del paciente:

Los pacientes que si acuden son atendidos y recetados inmediatamente.

Historial clínico:

Únicamente los pacientes tendrán acceso a su historial clínico de consulta, o de laboratorios mediante un usuario y contraseña, que se les es proporcionado por parte del médico.

Comunicación medico paciente:

La comunicación entre la Unidad Médica Ángel y los pacientes es muy importante ya que de ella dependen que se logren objetivos, para una mejor y pronta atención a los pacientes

CRONOGRAMA DE ACTIVIDADES

a) Nombre del proyecto

“Desarrollo de una aplicación Web para consulta de Resultados clínicos y servicios de la Unidad Médica Ángel.”

b) Objetivo del proyecto

Desarrollar una aplicación Web que permita consultar los servicios de la Unidad Médica Ángel, así como resultados clínicos y solicitud de citas.

c) Cronograma preliminar de actividades

ACTIVIDAD	SEMANAS																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
Análisis de requisitos	P	■															
	R																
Recopilación de información	P			■													
	R																
Diseño físico y lógico de la aplicación	P					■											
	R																
Diseño de la base de datos	P								■								
	R																
Prueba y corrección de errores	P										■						
	R																
Elaboración de manuales	P														■		
	R																
Implementación	P															■	
	R																
OBSERVACIONES		1er. Informe				2º. Informe				3er. Informe							
El reporte final: 13 y 14 de junio 2013		Febrero 25-26				Abril 08-09				Mayo 08-09							
ENTREGA DE REPORTES		Docente	M.A Jorge William Figueroa <i>Corzo</i>														
		Alumno	Alfredo Martínez Cruz														
		Jefe Dpto.	M.C. Roberto Cruz Gardillo														

DESCRIPCIÓN DETALLADA DE LAS ACTIVIDADES

- Análisis de requisitos para la implementación de la aplicación Web.
- Recopilación de Información.
- Diagramación de la interfaz
- Diseño físico de la aplicación Web.
- Diseño lógico de la aplicación web.
- Diseño de la base de datos.
- Prueba
- Corrección de errores.
- Elaboración de Manuales
- Implementación
- Mejoras

ALCANCES

Esta aplicación permite, en primer lugar a los médicos, dar a conocer vía internet, los expedientes de los pacientes de una forma personalizada y confidencial, así como dar recomendaciones, avisos urgentes, resultados de diagnósticos, etc., y en caso de ser necesario, solicitar una consulta en línea y la oportunidad de hacer citas personales en fechas posteriores. Así mismo, publicar anuncios y enlaces que puedan ser de utilidad a sus pacientes.

Permitirá al paciente, consultar el resultado de sus análisis y su expediente clínico, así como exteriorizar sus dudas al médico sin la necesidad de esperar su cita personal, podrá solicitar citas en línea y la oportunidad de elegir la fecha y horario de su conveniencia de acuerdo a las recomendaciones de su médico, y por lo tanto tener una mayor seguridad en el tratamiento de su salud.

LIMITACIONES

La administración de esta aplicación web es exclusiva del personal médico y administrativo de la Unidad Médica Ángel, y fuera de ellos, será de uso exclusivo para todos los pacientes de este centro médico, para consulta de resultados de análisis clínicos, consultas médicas, resolución de dudas y solicitud de citas médicas personales.

FUNDAMENTO TEÓRICO

APLICACIÓN WEB

1.1. ANTECEDENTES

Se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador.

Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, a la independencia del sistema operativo, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales. Existen aplicaciones como los webmails, wikis, weblogs, tiendas en línea y la propia Wikipedia que son ejemplos bien conocidos de aplicaciones web.

Es importante mencionar que una página Web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responderá a cada una de sus acciones, como por ejemplo rellenar y enviar formularios, participar en juegos diversos y acceder a gestores de base de datos de todo tipo.

1.2. INTERFAZ

Las interfaces web tienen ciertas limitaciones en las funcionalidades que se ofrecen al usuario. Hay funcionalidades comunes en las aplicaciones de escritorio como dibujar en la pantalla o arrastrar-y-soltar que no están soportadas por las tecnologías web estándar. Los desarrolladores web generalmente utilizan lenguajes interpretados (scripts) en el lado del cliente para añadir más funcionalidades, especialmente para ofrecer una experiencia interactiva que no requiera recargar la página cada vez (lo que suele resultar molesto a los usuarios). Recientemente se han desarrollado tecnologías para coordinar estos lenguajes con las tecnologías en el lado del servidor. Como ejemplo, AJAX es una técnica de desarrollo web que usa una combinación de varias tecnologías.

1.3. CONSIDERACIONES TÉCNICAS

Una ventaja significativa es que las aplicaciones web deberían funcionar igual e independientemente de la versión del sistema operativo instalado en el cliente. En vez de crear clientes para Windows, Mac OS X, GNU/Linux y otros sistemas operativos, la aplicación web se escribe una vez y se ejecuta igual en todas partes. Sin embargo, hay aplicaciones inconsistentes escritas con HTML, CSS, DOM y otras especificaciones estándar para navegadores web que pueden causar problemas en el desarrollo y soporte de estas aplicaciones, principalmente debido a la falta de adhesión de los navegadores a dichos estándares web (especialmente versiones de Internet Explorer anteriores a la 7.0). Adicionalmente, la posibilidad de los usuarios de personalizar muchas de las características de la interfaz (tamaño y color de fuentes, tipos de fuentes, inhabilitar Java script) puede interferir con la consistencia de la aplicación web.

Otra aproximación es utilizar Adobe Flash Player o Java applets para desarrollar parte o toda la interfaz de usuario. Como casi todos los navegadores incluyen soporte para estas tecnologías (usualmente por medio de plug-ins), las aplicaciones basadas en

Flash o Java pueden ser implementadas con aproximadamente la misma facilidad. Dado que ignoran las configuraciones de los navegadores, estas tecnologías permiten más control sobre la interfaz, aunque las incompatibilidades entre implementaciones Flash o Java puedan crear nuevas complicaciones, debido a que no son estándares. Por las similitudes con una arquitectura cliente-servidor, con un cliente "no ligero", existen discrepancias sobre el hecho de llamar a estos sistemas "aplicaciones web"; un término alternativo es "Aplicación Enriquecida de Internet".

1.4. ESTRUCTURA DE LAS APLICACIONES WEB

Aunque existen muchas variaciones posibles, una aplicación web está normalmente estructurada como una aplicación de tres-capas. En su forma más común, el navegador web ofrece la primera capa, y un motor capaz de usar alguna tecnología web dinámica (ejemplo: PHP, Java Servlets o ASP, ASP.NET, CGI, ColFusion, embPerl, Python (programminglanguage) (o Ruby on Rails) que constituye la capa intermedia. Por último, una base de datos constituye la tercera y última capa.

El navegador web manda peticiones a la capa intermedia que ofrece servicios valiéndose de consultas y actualizaciones a la base de datos y a su vez proporciona una interfaz de usuario

1.5. VENTAJAS

- **Portables:** Es independiente del ordenador donde se utilice (un PC de sobremesa, un portátil...) porque se accede a través de una página web (sólo es necesario disponer de acceso a Internet). La reciente tendencia al acceso a las aplicaciones web a través de teléfonos móviles requiere sin embargo un diseño específico de los ficheros [CSS](#) para no dificultar el acceso de estos usuarios.

- **La disponibilidad suele ser alta** porque el servicio se ofrece desde múltiples localizaciones para asegurar la continuidad del mismo.

- **Los virus no dañan** los datos porque éstos están guardados en el servidor de la aplicación.

- **Colaboración:** Gracias a que el acceso al servicio se realiza desde una única ubicación es sencillo el acceso y compartición de datos por parte de varios usuarios. Tiene mucho sentido, por ejemplo, en aplicaciones online de calendarios u oficina.

- Los navegadores ofrecen **cada vez más y mejores funcionalidades** para crear aplicaciones web ricas ([RIAs](#)).

- **Ahorra tiempo:** Se pueden realizar tareas sencillas sin necesidad de descargar ni instalar ningún programa.

- **No hay problemas de compatibilidad:** Basta tener un navegador actualizado para poder utilizarlas.

- **No ocupan espacio** en nuestro disco duro.

- **Actualizaciones inmediatas:** Como el software lo gestiona el propio desarrollador, cuando nos conectamos estamos usando siempre la última versión que haya lanzado.

- **Consumo de recursos bajo:** Dado que toda (o gran parte) de la aplicación no se encuentra en nuestro ordenador, muchas de las tareas que realiza el software no consumen recursos nuestros porque se realizan desde otro ordenador.

- **Multiplataforma:** Se pueden usar desde cualquier sistema operativo porque sólo es necesario tener un navegador.

1.6. INCONVENIENTES

- Habitualmente ofrecen **menos funcionalidades que las aplicaciones de escritorio**. Se debe a que las funcionalidades que se pueden realizar desde un navegador son más limitadas que las que se pueden realizar desde el sistema operativo. Pero cada vez los navegadores están más preparados para mejorar en este aspecto. La aparición de HTML 5 representa un hito en este sentido. Es posible añadir funcionalidades a estas aplicaciones gracias al uso de Aplicaciones de Internet Enriquecidas.

- **La disponibilidad depende de un tercero**, el proveedor de la conexión a internet o el que provee el enlace entre el servidor de la aplicación y el cliente. Así que la disponibilidad del servicio está supeditada al proveedor

1.7. DIFERENCIA ENTRE APLICACIÓN WEB Y APLICACIÓN DE INTERNET ENRIQUECIDA

Las aplicaciones web se ejecutan nativamente desde el navegador. Pero existen algunas aplicaciones que funcionan desde el navegador pero además requieren la instalación de un software en el ordenador para poder utilizarse. Estas aplicaciones se denominan Aplicaciones de Internet Ricas. El motivo de usar este software adicional es que hay muchas funcionalidades que los navegadores no pueden ofrecer, y él **enriquece a las aplicaciones web ofreciendo dichas funcionalidades**. .

Ejemplos de funcionalidades que pueden ofrecer los programas online gracias al uso de software instalado:

- Procesamiento de imágenes
- Captura de imágenes
- Uso de webcam / Captura de video

PROCEDIMIENTO Y DESCRIPCION DE ACTIVIDADES REALIZADAS

Existen numerosos lenguajes de programación empleados para el desarrollo de aplicaciones web en el servidor, entre los que destacan:

PHP

PHP es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página Web resultante. PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes. PHP puede ser usado en la mayoría de los servidores web al igual que en casi todos los sistemas operativos y plataformas sin ningún costo.

PHP fue creado originalmente por Rasmus Lerdorf en 1995. Actualmente el lenguaje sigue siendo desarrollado con nuevas funciones por el grupo PHP.¹ Este lenguaje forma parte del software libre publicado bajo la licencia PHP que es incompatible con la Licencia Pública General de GNU debido a las restricciones del uso del término *PHP*.²

JAVA con sus tecnologías Java Servlets y JavaServerPages (JSP).

El lenguaje de programación Java fue originalmente desarrollado por James Gosling de Sun Microsystems (la cual fue adquirida por la compañía Oracle) y publicado en el 1995 como un componente fundamental de la plataforma Java de Sun Microsystems. Su sintaxis deriva mucho de C y C++, pero tiene menos facilidades de bajo nivel que cualquiera de ellos. Las aplicaciones de Java son generalmente compiladas a bytecode (clase Java) que puede ejecutarse en

cualquier máquina virtual Java (JVM) sin importar la arquitectura de la computadora subyacente. Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos y basado en clases que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo (conocido en inglés como *WORA*, o "*write once, run anywhere*"), lo que quiere decir que el código que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra. Java es, a partir del 2012, uno de los lenguajes de programación más populares en uso, particularmente para aplicaciones de cliente-servidor de web, con unos 10 millones de usuarios reportados.^{1 2}

La compañía Sun desarrolló la implementación de referencia original para los compiladores de Java, máquinas virtuales, y librerías de clases en 1991 y las publicó por primera vez en el 1995. A partir de mayo del 2007, en cumplimiento con las especificaciones del Proceso de la Comunidad Java, Sun volvió a licenciar la mayoría de sus tecnologías de Java bajo la Licencia Pública General de GNU. Otros también han desarrollado implementaciones alternas a estas tecnologías de Sun, tales como el Compilador de Java de GNU y el GNU Classpath.

PERL

Perl es un lenguaje de programación diseñado por Larry Wall en 1987. Perl toma características del lenguaje C, del lenguaje interpretado bourneshell (sh), AWK, sed, Lisp y, en un grado inferior, de muchos otros lenguajes de programación.

Estructuralmente, Perl está basado en un estilo de bloques como los del C o AWK, y fue ampliamente adoptado por su destreza en el procesado de texto y no tener ninguna de las limitaciones de los otros lenguajes de script.

RUBY

Ruby es un lenguaje de programación interpretado, reflexivo y orientado a objetos, creado por el programador japonés Yukihiro "Matz" Matsumoto, quien comenzó a trabajar en Ruby en 1993, y lo presentó públicamente en 1995. Combina una sintaxis inspirada en Python y Perl con características de programación orientada a objetos similares a Smalltalk. Comparte también funcionalidad con otros lenguajes de programación como Lisp, Lua, Dylan y CLU. Ruby es un lenguaje de programación interpretado en una sola pasada y su implementación oficial es distribuida bajo una licencia de software libre.

PYTHON

Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis muy limpia y que favorezca un código legible.

Se trata de un lenguaje de programación multi paradigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, uso tipo dinámico y es multiplataforma.

Es administrado por la Python Software Foundation. Posee una licencia de código abierto, denominada Python Software Foundation License,¹ que es compatible con la Licencia pública general de GNU a partir de la versión 2.1.1, e incompatible en ciertas versiones anteriores.

C# Y VISUAL BASIC CON SUS TECNOLOGÍAS ASP/ASP.NET

Visual Basic es un lenguaje de programación dirigido por eventos, desarrollado por Alan Cooper para Microsoft. Este lenguaje de programación es un dialecto de BASIC, con importantes agregados. Su primera versión fue presentada en 1991, con la intención de simplificar la programación utilizando un ambiente de desarrollo completamente gráfico que facilitara la creación de interfaces gráficas y, en cierta medida, también la programación misma.

La última versión fue la 6, liberada en 1998, para la que Microsoft extendió el soporte hasta marzo de 2008.

En 2001 Microsoft propuso abandonar el desarrollo basado en la API Win32 y pasar a un framework o marco común de librerías, independiente de la versión del sistema operativo, .NET Framework, a través de Visual Basic .NET (y otros lenguajes como C Sharp (C#) de fácil transición de código entre ellos); fue el sucesor de Visual Basic 6.

Aunque Visual Basic es de propósito general, también provee facilidades para el desarrollo de aplicaciones de bases de datos usando Data Access Objects, Remote Data Objects o ActiveX Data Objects.

Visual Basic contiene un entorno de desarrollo integrado o IDE que integra editor de textos para edición del código fuente, un depurador, un compilador (y enlazador) y un editor de interfaces gráficas o GUI.

También son muy utilizados otros lenguajes o arquitecturas que no son propiamente lenguajes de programación, como HTML o XML.

Se utilizan para servir los datos adecuados a las necesidades del usuario, en función de cómo hayan sido definidos por el dueño de la aplicación. Los datos se almacenan en alguna base de datos estándar.

HTML

Siglas de **Híper Text Markup Language** («lenguaje de marcado híper textual»), hace referencia al lenguaje de marcado predominante para la elaboración de páginas web que se utiliza para describir y traducir la estructura y la información en forma de texto, así como para complementar el texto con objetos tales como imágenes. El HTML se escribe en forma de «etiquetas», rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un *script* (por ejemplo, JavaScript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

HTML también sirve para referirse al contenido del tipo de MIME text/html o todavía más ampliamente como un término genérico para el HTML, ya sea en forma descendida del XML (como XHTML 1.0 y posteriores) o en forma descendida directamente de SGML (como HTML 4.01 y anteriores).

ELEMENTOS

Los elementos son la estructura básica de HTML. Los elementos tienen dos propiedades básicas: atributos y contenido. Cada atributo y contenido tiene ciertas restricciones para que se considere válido al documento HTML. Un elemento generalmente tiene una etiqueta de inicio (por ejemplo, `<nombre-de-elemento>`) y una etiqueta de cierre (por ejemplo, `</nombre-de-elemento>`). Los atributos del elemento están contenidos en la etiqueta de inicio y el contenido está ubicado entre las dos etiquetas (por ejemplo, `<nombre-de-elemento atributo="valor">Contenido</nombre-de-elemento>`). Algunos elementos, tales como `
`, no tienen contenido ni llevan una etiqueta de cierre. Debajo se listan varios tipos de elementos de marcado usados en HTML.

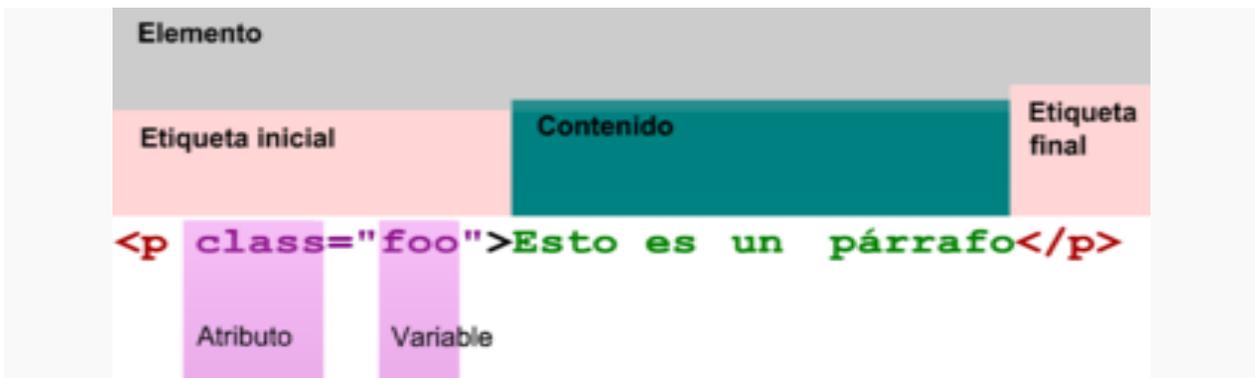


FIGURA 1. Estructura general de una línea de código

El marcado *estructural* describe el propósito del texto. Por ejemplo, `<h2>Golf</h2>` establece «Golf» como un encabezamiento de segundo nivel, el cual se mostraría en un navegador de una manera similar al título «Marcado HTML» al principio de esta sección. El marcado estructural no define cómo se verá el elemento, pero la mayoría de los navegadores web han estandarizado el formato de los elementos. Puede aplicarse un formato específico al texto por medio de hojas de estilo en cascada.

El marcado *presentacional* describe la apariencia del texto, sin importar su función. Por ejemplo, `negrita` indica que los navegadores web visuales deben mostrar el texto en **negrita**, pero no indica qué deben hacer los navegadores web que muestran el contenido de otra manera (por ejemplo, los que leen el texto en voz alta). En el caso de `negrita` e `<i>itálica</i>`, existen elementos que se ven de la misma manera pero tienen una naturaleza más semántica: `énfasis fuerte` y `énfasis`. Es fácil ver cómo un lector de pantalla debería interpretar estos dos elementos. Sin embargo, son equivalentes a sus correspondientes elementos presentacionales: un lector de pantalla no debería decir más fuerte el nombre de un libro, aunque éste esté en *itálicas* en una pantalla. La mayoría del marcado presentacional ha sido desechada con HTML 4.0, en favor de hojas de estilo en cascada.

El marcado *híper textual* se utiliza para enlazar partes del documento con otros documentos o con otras partes del mismo documento. Para crear un enlace es necesario utilizar la etiqueta de ancla `<a>` junto con el atributo `href`, que establecerá la dirección URL a la que apunta el enlace. Por ejemplo, un enlace que muestre el texto de la dirección y vaya hacia nuestra Wikipedia podría ser de la forma `http://www.wikipedia.org`. También se pueden crear enlaces sobre otros objetos, tales como imágenes ``.

ATRIBUTOS

La mayoría de los atributos de un elemento son pares nombre-valor, separados por un signo de igual «=» y escritos en la etiqueta de comienzo de un elemento, después del nombre de éste. El valor puede estar rodeado por comillas dobles o simples, aunque ciertos tipos de valores pueden estar sin comillas en HTML (pero no en XHTML).^{8 9} De todas maneras, dejar los valores sin comillas es considerado poco seguro.¹⁰ En contraste con los pares nombre-elemento, hay algunos atributos que afectan al elemento simplemente por su presencia¹¹ (tal como el atributo `ismappara` el elemento `img`).

XML

Siglas en inglés de *eXtensible Markup Language* ('lenguaje de marcas extensible'), es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C). Deriva del lenguaje SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML) para estructurar documentos grandes. A diferencia de otros lenguajes, XML da soporte a bases de datos, siendo útil cuando varias aplicaciones se deben comunicar entre sí o integrar información. (Bases de datos Silberschatz).

XML no ha nacido sólo para su aplicación para Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo y casi cualquier cosa imaginable.

XML es una tecnología sencilla que tiene a su alrededor otras que la complementan y la hacen mucho más grande y con unas posibilidades mucho mayores. Tiene un papel muy importante en la actualidad ya que permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil.

BASE DE DATOS



El sistema de base de datos operacional MySQL es hoy en día uno de los más importantes en lo que hace al diseño y programación de base de datos de tipo relacional. Cuenta con millones de aplicaciones y aparece en el mundo informático como una de las más utilizadas por usuarios del medio. El programa MySQL se usa como servidor a través del cual pueden conectarse múltiples usuarios y utilizarlo al mismo tiempo.

La historia del MySQL (cuya sigla en inglés se traslada a My Structured Query Language o Lenguaje de Consulta Estructurado) se remite a principios de la década de 1980. Programadores de IBM lo desarrollaron para contar con un código de programación que permitiera generar múltiples y extendidas bases de datos para empresas y organizaciones de diferente tipo. Desde esta época numerosas versiones han surgido y muchas de ellas fueron de gran importancia. Hoy en día MySQL es desarrollado por la empresa Sun Microsystems.

ABC: <http://www.definicionabc.com/tecnologia/mysql.php#ixzz2WQezJY7I>.

Las bases de datos pueden clasificarse de varias maneras, de acuerdo al contexto que se esté manejando, la utilidad de las mismas o las necesidades que satisfagan.

Según la variabilidad de los datos almacenados

BASES DE DATOS ESTÁTICAS

Son bases de datos de sólo lectura, utilizadas primordialmente para almacenar datos históricos que posteriormente se pueden utilizar para estudiar el comportamiento de un conjunto de datos a través del tiempo, realizar proyecciones, tomar decisiones y realizar análisis de datos para inteligencia empresarial.

BASES DE DATOS DINÁMICAS

Éstas son bases de datos donde la información almacenada se modifica con el tiempo, permitiendo operaciones como actualización, borrado y adición de datos, además de las operaciones fundamentales de consulta. Un ejemplo de esto puede ser la base de datos utilizada en un sistema de información de un supermercado, una farmacia, un videoclub o una empresa.

BASES DE DATOS BIBLIOGRÁFICAS

Sólo contienen un subrogante (representante) de la fuente primaria, que permite localizarla. Un registro típico de una base de datos bibliográfica contiene información sobre el autor, fecha de publicación, editorial, título, edición, de una determinada publicación, etc. Puede contener un resumen o extracto de la publicación original, pero nunca el texto completo, porque si no, estaríamos en presencia de una base de datos a texto completo (o de fuentes primarias —ver más abajo). Como su nombre lo indica, el contenido son cifras o números. Por ejemplo, una colección de resultados de análisis de laboratorio, entre otras.

BASES DE DATOS DE TEXTO COMPLETO

Almacenan las fuentes primarias, como por ejemplo, todo el contenido de todas las ediciones de una colección de revistas científicas.

MODELOS DE BASE DE DATOS.

Además de la clasificación por la función de las bases de datos, éstas también se pueden clasificar de acuerdo a su modelo de administración de datos.

Un modelo de datos es básicamente una "descripción" de algo conocido como *contenedor de datos* (algo en donde se guarda la información), así como de los métodos para almacenar y recuperar información de esos contenedores. Los modelos de datos no son cosas físicas: son abstracciones que permiten la implementación de un sistema eficiente de *base de datos*; por lo general se refieren a algoritmos, y conceptos matemáticos.

Algunos modelos con frecuencia utilizados en las bases de datos:

BASES DE DATOS JERÁRQUICAS.

Artículo principal: Base de datos jerárquica.

En este modelo los datos se organizan en forma de árbol invertido (algunos dicen raíz), en donde un *nodo padre* de información puede tener varios *hijos*. El nodo que no tiene padres es llamado *raíz*, y a los nodos que no tienen hijos se los conoce como *hojas*.

Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento.

Una de las principales limitaciones de este modelo es su incapacidad de representar eficientemente la redundancia de datos.

BASE DE DATOS DE RED.

Artículo principal: Base de datos de red.

Éste es un modelo ligeramente distinto del jerárquico; su diferencia fundamental es la modificación del concepto de *nodo*: se permite que un mismo nodo tenga varios padres (posibilidad no permitida en el modelo jerárquico).

Fue una gran mejora con respecto al modelo jerárquico, ya que ofrecía una solución eficiente al problema de redundancia de datos; pero, aun así, la dificultad que significa administrar la información en una base de datos de red ha significado que sea un modelo utilizado en su mayoría por programadores más que por usuarios finales.

BASES DE DATOS TRANSACCIONALES.

Son bases de datos cuyo único fin es el envío y recepción de datos a grandes velocidades, estas bases son muy poco comunes y están dirigidas por lo general al entorno de análisis de calidad, datos de producción e industrial, es importante entender que su fin único es recolectar y recuperar los datos a la mayor velocidad posible, por lo tanto la redundancia y duplicación de información no es un problema como con las demás bases de datos, por lo general para poderlas aprovechar al máximo permiten algún tipo de conectividad a bases de datos relacionales.

Un ejemplo habitual de transacción es el traspaso de una cantidad de dinero entre cuentas bancarias. Normalmente se realiza mediante dos operaciones distintas, una en la que se decrementa el saldo de la cuenta origen y otra en la que incrementamos el saldo de la cuenta destino. Para garantizar la atomicidad del sistema (es decir, para que no aparezca o desaparezca dinero), las dos operaciones deben ser atómicas, es decir, el sistema debe garantizar que, bajo cualquier circunstancia (incluso una caída del sistema), el resultado final es que, o bien se han realizado las dos operaciones, o bien no se ha realizado ninguna.

BASES DE DATOS RELACIONALES.

Artículo principal: Modelo relacional.

Artículo principal: Base de datos relacional.

Éste es el modelo utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente. Tras ser postulados sus fundamentos en 1970 por Edgar Frank Codd, de los laboratorios IBM en San José (California), no tardó en consolidarse como un nuevo paradigma en los modelos de base de datos. Su idea fundamental es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos llamados "tuplas". Pese a que ésta es la teoría de las bases de datos relacionales creadas por Codd, la mayoría de las veces se conceptualiza de una manera más fácil de imaginar. Esto es pensando en cada relación como si fuese una tabla que está compuesta por *registros* (las filas de una tabla), que representarían las tuplas, y *campos* (las columnas de una tabla).

En este modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos. La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

El lenguaje más habitual para construir las consultas a bases de datos relacionales es SQL, *Structured Query Language* o *Lenguaje Estructurado de Consultas*, un estándar implementado por los principales motores o sistemas de gestión de bases de datos relacionales.

Durante su diseño, una base de datos relacional pasa por un proceso al que se le conoce como normalización de una base de datos.

Durante los años 80 la aparición de BASE produjo una revolución en los lenguajes de programación y sistemas de administración de datos. Aunque nunca debe olvidarse que Base no utilizaba SQL como lenguaje base para su gestión.

BASES DE DATOS MULTIDIMENSIONALES.

Artículo principal: Base de datos multidimensional.

Son bases de datos ideadas para desarrollar aplicaciones muy concretas, como creación de **Cubos OLAP**. Básicamente no se diferencian demasiado de las bases de datos relacionales (una tabla en una base de datos relacional podría serlo también en una base de datos multidimensional), la diferencia está más bien a nivel conceptual; en las bases de datos multidimensionales los campos o atributos de una tabla pueden ser de dos tipos, o bien representan dimensiones de la tabla, o bien representan métricas que se desean estudiar.

BASES DE DATOS ORIENTADAS A OBJETOS.

Artículo principal: Base de datos orientada a objetos.

Este modelo, bastante reciente, y propio de los modelos informáticos orientados a objetos, trata de almacenar en la base de datos los *objetos* completos (estado y comportamiento).

Una base de datos orientada a objetos es una base de datos que incorpora todos los conceptos importantes del paradigma de objetos:

- Encapsulación - Propiedad que permite ocultar la información al resto de los objetos, impidiendo así accesos incorrectos o conflictos.
- Herencia - Propiedad a través de la cual los objetos heredan comportamiento dentro de una jerarquía de clases.
- Polimorfismo - Propiedad de una operación mediante la cual puede ser aplicada a distintos tipos de objetos.

En bases de datos orientadas a objetos, los usuarios pueden definir operaciones sobre los datos como parte de la definición de la base de datos. Una operación (llamada función) se especifica en dos partes. La interfaz (o signatura) de una operación incluye el nombre de la operación y los tipos de datos de sus argumentos (o parámetros). La implementación (o método) de la operación se especifica separadamente y puede modificarse sin afectar la interfaz. Los programas de aplicación de los usuarios pueden operar sobre los datos invocando a dichas operaciones a través de sus nombres y argumentos, sea cual sea la forma en la que se han implementado. Esto podría denominarse independencia entre programas y operaciones.

SQL:2003, es el estándar de SQL92 ampliado, soporta los conceptos orientados a objetos y mantiene la compatibilidad con SQL92.

CONCLUSIÓN

Gracias a su implementación, se logró una optimización en los recursos de la Unidad Médica Ángel, logrando una mejor calidad en sus servicios y satisfacción por parte de los pacientes, quienes ahora reciben una mejor atención, tienen mayor seguridad al poder acceder de manera confidencial a sus expedientes médicos, y cuentan con la facilidad de mantener una constante comunicación con sus médicos de cabecera, para atender cualquier duda o situación urgente de manera rápida y eficaz, procurando siempre, en primer lugar, la correcta atención a sus necesidades médicas.

Me he permitido antes que nada, contribuir de manera eficiente a la sociedad, brindándoles un producto informático que permita el desarrollo de la empresa muestra, el mismo para una posible aplicación en otras áreas que la requieran, contribuyendo de esta manera al desarrollo de aplicaciones informáticas para un servicio más eficiente de las empresas del estado, logrando así, un pronto desarrollo tecnológico del mismo.

RECOMENDACIONES

Se exhorta a la Unidad Médica Ángel, el uso constante de la aplicación proporcionada, recomendando que no se limiten en los servicios que pueden proporcionar a sus pacientes por medio de esta, desde el cambio constante en el diseño de su presentación visual, hasta la cantidad de innovaciones que le pueden agregar en materia de servicios, información y enlaces que permitan lograr una mayor satisfacción de los clientes y una mejora continua en la implementación de la aplicación.

BIBLIOGRAFÍA

- Alfredo Espinosa R., B. M. (junio de 2007). *Instituto de Investigaciones Electricas*. Recuperado el 9 de Mayo de 2013, de Instituto de Investigaciones Electricas: <http://www.iie.org.mx/boletin022007/apli2.pdf>
- Correa, L. A. (22 de Noviembre de 2006). *Desarrollo Web*. Recuperado el 6 de Junio de 2013, de Desarrollo Web: <http://www.desarrolloweb.com/articulos/aplicaciones-web-usabilidad-practica-exito.html>
- Estructura y programacion. (s.f.). *Estructura y programacion*. Recuperado el 2 de Mayo de 2013, de Estructura y programacion: <http://www.estructurayprogramacion.com/materias/programacion-web/tecnologias-para-el-desarrollo-de-aplicaciones-web/>
- Gutiérrez, J. S. (7 de Mayo de 2007). *El Mundo Informático*. Recuperado el 5 de Mayo de 2013, de El Mundo Informático: <http://jorgesaavedra.wordpress.com/2007/05/05/lenguajes-de-programacion/>
- Luis A. Guerrero, R. C. (s.f.). *Pontificia Universidad Católica de Chile*. Recuperado el 19 de Mayo de 2013, de Pontificia Universidad Católica de Chile: <http://users.dcc.uchile.cl/~luguerre/papers/CLEI-98/>
- Marcos, P. C. (s.f.). *Universidad Rey Juan Carlos*. Recuperado el 3 de Junio de 2013, de Universidad Rey Juan Carlos: <http://www.dlsi.ua.es/webe01/articulos/s112.pdf>
- Mateu, C. (s.f.). *Universidad Oberta de Catalunya*. Recuperado el 21 de abril de 2013, de Universidad Oberta de Catalunya: http://www.sw-computacion.f2s.com/Linux/004-Desarrollo_de_aplicaciones_web.pdf
- Pablo Castells. (27 de Marzo de 2012). *Universidad Autónoma de Madrid*. Recuperado el 2013 de Abril de 16, de Universidad Autónoma de Madrid: <http://arantxa.ii.uam.es/~castells/docencia/poo/6-guis.pdf>
- Paco Blanco, J. C. (2009). *Universidad Autónoma de Madrid*. Recuperado el 23 de Mayo de 2013, de Universidad Autónoma de Madrid:

http://www.adamwesterski.com/wp-content/files/docsCursos/Agile_doc_TemasAnv.pdf

Paredes, C. (Septiembre de 2009). *Universidad Católica Nuestra Señora de la Asunción*. Recuperado el 2 de Junio de 2013, de Universidad Católica Nuestra Señora de la Asunción: http://www.jeuazarru.com/docs/Internet_apps_para_escritorio.pdf

Pedro Juan Molina Moreno. (Marzo de 2003). *Universidad Politécnica de Valencia*. Recuperado el 16 de Abril de 2013, de Universidad Politécnica de Valencia: <http://pjmolina.com/papers/TesisPjmolina.pdf>

Rafael Barzanallana. (8 de Agosto de 2012). *Universidad de Murcia*. Recuperado el 14 de Abril de 2013, de Universidad de Murcia: <http://www.um.es/docencia/barzana/DIVULGACION/INFORMATICA/Historia-desarrollo-aplicaciones-web.html>