

DESCRIPCIÓN DE LA EMPRESA

Nombre: UNIVERSAL DEVELOPMENT

Giro: Diseño y desarrollo de aplicaciones de software

Domicilio: Blvd. Juan Pablo II 1302. Fracc. Exhacienda La Cantera, C.P. 20200, Aguascalientes, Ags.

Teléfono: 449 910 5000

Correo electrónico: universal-dp@gmail.com

Página web: www.univeresal-dp.com.mx

Antecedentes históricos:

UNIVERSAL DEVELOPMENT es una empresa 100% mexicana con más de 20 años de experiencia en desarrollo de software, fundada en 1996 en la ciudad de Aguascalientes por el Ing. Felipe de Jesús Velasco Nieto, un visionario que se propuso construir una empresa dedicada al mejor diseño y desarrollo de aplicaciones informáticas. Velasco concibió un nuevo tipo de automatización de flujo de trabajo: una suite de software completa que integraba firmemente toda la funcionalidad necesaria y era intuitivo y accesible para los usuarios no técnicos.

Misión:

Brindar asesoramiento informático, para empresas o usuarios, con personal capacitado y experimentado, para que las empresas, diseñadores, y usuarios tengan productos de fácil acceso y de una buena calidad, que tengan un mejor conocimiento informático gracias a nuestras asesorías.

Ser una empresa dedicada a la comercialización, y distribución de productos informáticos, desarrollo de aplicaciones informáticas de gestión, servicio técnico, ofreciendo una solución global para nuestros clientes.

Visión:

Para el año 2020 UNIVERSAL DEVELOPMENT busca:

- Ser reconocida como una de las diez principales empresas en la mejora del campo de venta de los servicios de desarrollo de software en México y Latinoamérica.
- Que nuestros empleados adquieran una mayor experiencia en sus respectivas áreas de especialización.
- Ser personas sumamente responsables y capacitadas para lograr una buena imagen e impacto de confianza en los clientes, empresas o microempresas con sistemas manuales que se puedan automatizar.
- Ser una empresa en constante innovación, ya que el sector de la tecnología así lo requiere, provocado por los avances tecnológicos.

Todo esto por medio de capacitaciones periódicas en lenguajes de programación, nuevas tecnologías, metodologías y calidad.

Valores:

Nuestros valores sobre los que se sostiene esta estrategia de crecimiento y diferenciación en el servicio son:

- Trabajo en equipo: promoviendo y apoyando un equipo homogéneo polivalente e interdepartamental.
- Colaboración: nos integramos con nuestros proveedores y clientes, para mejorar día a día la calidad de los mismos para satisfacer sus necesidades.
- Servicio: cumplimos con nuestros compromisos y nos hacemos responsables de nuestro rendimiento en todas nuestras decisiones y acciones, basándonos en una gran voluntad de servicio, por y para nuestros clientes.
- Innovación y mejora continua: nos damos cuenta de la importancia de mirar hacia el futuro, por tanto, ofrecemos lo último del mercado para dar un apoyo y servicio óptimo a nuestros clientes.
- Transparencia: la implicación y compromiso del personal no sería posible sin una absoluta transparencia en los procesos, disponiendo el personal de la máxima información de la empresa.
- Comunicación: proveemos y facilitamos la comunicación entre todos los niveles de la organización, disponiendo de herramientas eficaces, convocando los foros adecuados y con el compromiso constante de la dirección.
- Integridad y ética: promovemos un compromiso social y cumplimos nuestra normativa interna.
- Modelo de dirección participativa: el personal de la empresa asume responsabilidades y participa en el proceso de toma de decisiones.
- Formación: la empresa se preocupa de la formación continua en todos los ámbitos.
- Responsabilidad social corporativa: contribuimos activamente al mejoramiento social, económico y ambiental, para mejorar nuestra situación competitiva y valorativa.

Análisis FODA

Fortalezas:

- Calidad de nuestro producto
- Excelente forma de atención al cliente
- Seguridad en la infraestructura de la información
- Servicios de elevado nivel
- Empleados motivados

Oportunidades:

- Crecimiento fuerte en la industria
- Alta demanda de productos relacionados
- Bajas tarifas
- Estabilidad de la industria

Debilidades:

- Salarios inferiores a la media
- Problemas financieros
- Poca información del personal
- No tener licencias de los programas utilizados
- Desarrollo de software a una sola metodología

Amenazas:

- Elevada competencia
- Bajo potencial de crecimiento de nuestra empresa
- Aumento de precios
- Mala distribución del producto
- Perder seguridad en información

ACTA DE INICIO DE PROYECTO

NOMBRE DEL PROYECTO: MEDITEC

FECHA: 29/05/2019

Empresa / Organización	Nombre de la empresa u organización que va a desarrollar el software
Proyecto	MEDITEC
Fecha de preparación	29/05/2019
Cliente	Felipe De Jesús Velasco Nieto
Patrocinador principal	Felipe De Jesús Velasco Nieto
Gerente del proyecto	Arizbeth Santillán Tafoya

PROPÓSITO Y JUSTIFICACIÓN:

Desarrollar una aplicación de escritorio y web, para administrar todas las tareas de una clínica donde podemos llevar a cabo una sistematización para tener un mejor control de la información de los expedientes clínicos y reducir el costo en papelería y tiempo.

Actualmente no se cuenta con un sistema para la administración de la información de clínica teniendo el riesgo de perder información importante del cliente.

OBJETIVO:

Desarrollar una aplicación que sea capaz de realizar altas, bajas y cambios de la información relevante de la clínica, además de poder realizar citas en la misma y en la página web del consultorio.

ALCANCE:

- Una aplicación que tenga un apartado de expedientes en el cual realicen altas de los mismos que se tienen en físico y agregar los nuevos, además poder modificarlos, si así el doctor lo desea y por último consultarlos para poder darle continuidad al paciente.
- Se quiere que la aplicación sea segura y solo tengan acceso a los expedientes los médicos de acuerdo a su respectiva área ya que se tienen diferentes especializaciones, cada doctor tendrá su propio usuario y contraseña con seguridad media además se debe evitar la inyección de código.
- Incluir una interfaz de citas y llevar un control con los horarios para lograr citas únicas, implementando un sistema de consulta que señale disponibilidad en la fecha y horario de la cita.

IDENTIFICACIÓN DE RIESGOS

RIESGOS GENERALES	
Título	Descripción del riesgo
No hacer el login primero	Dejar el login hasta el último, produciendo más problemas por tener las demás interfaces funcionando excepto esta.
Problemas con la programación	Al momento de depurar el programa, el programador se moleste y se sature.
Errores en la base de datos	A la hora de querer insertar algún dato las relaciones no lo permitan y cause efectos negativos en la actitud del equipo.
Entregables de betas	A la hora de entregar la primera beta no sea lo que el cliente esperaba, y como consecuencia, el líder de proyecto reaccione de manera errónea contra el equipo.
Diseñador	El diseñador no agregue los componentes correctos de la interfaz y se retrase el trabajo en el resto de las etapas de desarrollo.
Cliente	El cliente solicite más requerimientos de los acordados y estos estén fuera de alcance y presupuesto.

LISTA DE INTERESADOS (STAKEHOLDERS)

NOMBRE	CARGO	DEPARTAMENTO O DIVISIÓN
Marco Herrera	Dentista	Odontólogo
Antonio Hernández	Médico general	Medicina Familiar
Daniel Tolentino	Psicólogo	Psicología
Marisol De la Torre	Secretaria	Recepción

CRONOGRAMA DE HITOS PRINCIPALES

HITO	FECHA LIMITE
Entrevista con el cliente	29/05/2019
Requerimientos de sistema	04/06/2019
Desarrollo de diagramas	20/06/2019
Diseño de GUI	30/06/2019
Desarrollo	19/07/2019
Pruebas	20/07/2019
Documentación final	21/07/2019
Implementación	22/07/2019

ACTORES INVOLUCRADOS EN EL DESARROLLO DEL PROYECTO

NOMBRE DEL ACTOR	DESCRIPCIÓN DE ACTIVIDADES
Arizbeth Santillán Tafoya	Gerente del proyecto, Líder del proyecto
Alexia Escamilla Vega	Diseñador, Diseñador de base de datos

Tania Marisol de la Torre Monroy	Responsable de calidad
Marco Antonio Herrera Salas	Ingeniero de software, Administración de Base de Datos, Administrador de la configuración del proyecto
Carlos Daniel Herrera Tolentino	Responsable de pruebas, Analista de sistemas
Felipe De Jesús Velasco Nieto	Cliente

Nombre y firma del cliente

Nombre y firma del gerente

DESCRIPCIÓN DEL PROBLEMA

Descripción del problema

Actualmente en el consultorio médico el control de citas, pacientes, expedientes u horarios de los médicos, se lleva a cabo de forma manual por medio de papel, lo cual hace que sea lenta e imprecisa la atención y administración del establecimiento, además del excesivo consumo de papelería que el proceso genera.

Objetivos del proyecto

Objetivo general del sistema

Realizar un sistema informático para el consultorio, que sea capaz de administrar la agenda de los doctores, teniendo en cuenta la especialidad de cada uno para asignar el paciente adecuado a su consulta correspondiente. Que pueda automatizar la captura de expedientes para cada médico considerando el método actual de captura de información para facilitar su uso. Y, por último, que facilite al paciente la acción de agendar una cita en el consultorio, proporcionándole una vista en Google Maps de la ubicación del establecimiento e información general del consultorio.

Objetivos específicos

- Automatizar altas, bajas, consultas y cambios de expedientes de pacientes para medicina familiar.
- Automatizar altas, bajas, consultas y cambios de expedientes de pacientes para psicología.
- Automatizar altas, bajas, consultas y cambios de citas de pacientes.

Requisitos del proyecto

- Entrevista con el cliente para conocer sus necesidades y problemas, así conocer su presupuesto para establecer el alcance del proyecto.
- Mostrar avances al cliente para que autorice los cambios, si él considera que son oportunos o hace sugerencia de correcciones de acuerdo con sus necesidades.
- Mostrar prototipos al cliente de la interfaz para que apruebe los cambios de acuerdo con sus necesidades.
- El cliente debe dar la aprobación del documento para poder iniciar con el proyecto donde notificamos el costo y tiempo de entrega.
- Automatizar el registro de pacientes.
- Automatizar el registro de citas al consultorio y para cada médico.
- Automatizar y dar mayor seguridad a la elaboración de expedientes.
- Facilitar para el paciente y médicos el control de citas.

Requisitos del producto

- Desarrollador de programación C#
- Base de datos: SQL Server y MySQL.
- Servidor local con XAMPP.
- Para la página web: HTML, PHP, JavaScript, CSS.
- Para documentación: paquetería office.

Cronograma

[illegible]

Planificación de actividades

HITO	FECHA LÍMITE
Entrevista con el cliente para obtener información del consultorio como.	29/05/2019
Desglosar las preguntas para determinar los requerimiento funcionales y los requerimientos no funcionales	04/06/2019
Desarrollo de diagramas de casos de uso, clases, secuencia, estado, componentes, despliegue.	20/06/2019
Diseño de Interfaz de usuario con los módulos de inicio de sesión, menú, interfaz de ABCC, además la interfaz de reportes y por último la del ABCC de citas.	30/06/2019
Desarrollo de las bases de datos en MySQL y SQLSERVER para la gestión de la información y desarrollo de código de los módulos antes diseñados.	19/07/2019
Realización de pruebas para determinar la vulnerabilidad del código y corrección de errores.	20/07/2019
Documentación final del proyecto donde se presente toda información de planeación, diseño de diagramas y manual de usuario para ofrecer una guía al usuario al utilizar el sistema.	21/07/2019
Implementación del sistema en el consultorio con las conexiones necesarias de red, sus direcciones estáticas y la instalación del sistema con las bases de datos necesarias.	22/07/2019

Justificación

Se busca cubrir las necesidades de los pacientes, entre las cuales se encuentran: evitar una fila para ser atendido y después tenga que esperar para realizar la asignación de su próxima cita, lo más importante es lograr todo esto en un tiempo corto de espera. Se creó por las necesidades de los pacientes porque ellos necesitan de esto, de algo innovador.

Se pretende que el paciente desde la comodidad de su casa pueda realizar una cita, evitando una larga fila de espera directamente en recepción, se quiere evitar que el cliente descuide sus deberes por la necesidad de ir a re-agendar una nueva cita además de que se busca reducir la aglomeración en la clínica.

Se busca llevar a cabo la sistematización para tener un mejor control de la información personal de cada uno de los expedientes clínicos y reducir el costo en papelería, además de llevar información personal que no podemos perder ni combinar con otro expediente, este sistema nos ayudará a la mejora en cuestión de guardar la información y no perder tanto tiempo en el momento de buscar un expediente, dicho sistema se encargará de buscar el expediente solicitado de manera fluida, concisa y rápida.

Enunciado del alcance

Una aplicación que tenga un apartado de expedientes en el cual realicen altas de los mismos que se tienen en físico y agregar los nuevos, además poder modificarlos, si así el doctor lo desea, y por último consultarlos para poder darle continuidad al paciente.

Se quiere que la aplicación sea segura y solo tengan acceso a los expedientes los médicos de acuerdo a su respectiva área ya que se tienen diferentes especializaciones, cada doctor tendrá su propio usuario y contraseña con seguridad media además se debe evitar la inyección de código.

Incluir una interfaz de citas y llevar un control con los horarios para lograr citas únicas, implementando un sistema de consulta que señale disponibilidad en la fecha y horario de la cita.

Metodología

Entrevista con el cliente: es de suma importancia para el desarrollo del producto, debido a que en esta etapa se revelan mucha información a detalle de los requerimientos del cliente, sus inquietudes y expectativas. Las preguntas que se plantearán durante la entrevista son de acuerdo a los escenarios del consultorio, por ejemplo, a la recepcionista le preguntaremos qué actividades desempeña, herramientas que necesita para desempeñar su trabajo, etcétera. A los médicos les preguntaremos qué información maneja en sus formularios, por cuánto tiempo debe archivar la información. Y en general se

preguntarán horarios de consulta, tiempos de consulta y dudas que puedan ir surgiendo durante la entrevista.

Requerimientos de sistema: definirlos significa para nosotros el punto de partida del desarrollo del producto y para el cliente pinta una panorámica de lo que se espera al concluir. Los requerimientos que queremos obtener son: funcionales y no funcionales. Nos gustaría trabajar con la métrica FURPS para los no funcionales.

Desarrollo de diagramas: proporciona una idea clara de lo que debe ser y hacer el sistema. Es una guía también para todos los desarrolladores involucrados. En el desarrollo de diagramas comenzaremos por el de casos de uso para ilustrar al cliente de la funcionalidad del sistema y acordar en la recolección correcta de los requerimientos. Para el equipo de desarrollo usaremos diagramas de secuencia, de estado, de clases y E-R.

Diseño de GUI: es una vista incluso más clara del producto final del sistema. Se trata de la comunicación entre el usuario final con el software a desarrollar. De acuerdo a los requerimientos determinaremos los módulos a integrar del sistema y por medio de ello le daremos una vista al usuario de acuerdo a su formulario.

Desarrollo: de esta etapa depende el éxito del producto. Aquí se implementan todos los requerimientos y diagramas definidos con anterioridad. De acuerdo a las interfaces GUI continuamos con la etapa de codificación en donde interpretaremos la funcionalidad de los formularios en su forma manual a su forma automática, es decir realizaremos la automatización de actividades del consultorio.

Pruebas: para entregar un producto de calidad es necesario hacer la mayor cantidad de pruebas posibles, es responsabilidad del equipo de desarrollo que en las pruebas se solucionen todas las situaciones de error que le queremos evitar al cliente. De acuerdo a los errores que se presenten durante esta etapa consideraremos corregir cada uno de ellos de la forma que nosotros creamos que es la apropiada. Los errores comunes que se pueden presentar son: en la validación de usuarios, inyecciones a la base de datos.

Documentación final: es guía para el equipo de desarrollo, pero también se debe incluir una guía para el usuario. Teniendo en cuenta para cada caso el tipo de jerga que se debe utilizar. Es muy importante que el usuario sepa utilizar el sistema de forma adecuada, para que, de esta forma, las funcionalidades del sistema cumplan con sus expectativas.

Implementación: presentar al cliente el producto final en su equipo de cómputo para que pueda usarlo, es para nosotros el momento crítico del proyecto ya que ahí culminan los esfuerzos realizados. Para ello es necesario la evaluación de los escenarios, para conocer los componentes del software y hardware del usuario para adaptarlos a nuestro sistema y que este pueda funcionar adecuadamente.

MARCO TEÓRICO



HTML

“HTML es un lenguaje de marcado que se utiliza para el desarrollo de páginas de Internet. Se trata de las siglas que corresponden a HyperText Markup Language, es decir, Lenguaje de Marcas de Hipertexto”.

No obstante, este tipo de definiciones no nos dice mucho porque la definición es técnica. Para algunas personas al leer esto, piensan que HTML incluye el diseño gráfico de las páginas web, sin embargo, eso no es cierto ya que HTML sólo sirve para indicar como va ordenado el contenido de una página web. Esto lo hace por medio de las marcas de hipertexto las cuales son etiquetas conocidas en inglés como tags.

Hasta el momento ya tenemos una idea sobre lo que hace HTML, pero aún no sabemos cómo funcionan las etiquetas. Para esto vamos a imaginar que tienen que hacer un ensayo sobre algún tema, consideraremos que el ensayo será nuestra página web y la manera en la que va ordenado un ensayo es por un título, la introducción, el desarrollo y la conclusión, si se tiene en cuenta que esto nos dice cómo debe ir nuestro ensayo podemos considerar que éstas serían las etiquetas porque nos dicen la estructura que lleva, es decir, la forma en que se acomoda el texto en nuestro ensayo. Cabe mencionar que las etiquetas no sólo sirven para ordenar nuestro contenido, sino que ayudan a los buscadores como Google, Yahoo search y muchos otros a encontrar la información por medio de las etiquetas.

Ahora que entendemos el concepto de etiquetas, necesitamos poder identificarlas. Éstas se caracterizan porque van dentro de los caracteres menor que < y mayor que >, como en este ejemplo: <Aquí dentro va el nombre de la etiqueta>

Aunque eso no es todo debido a que deben tener un inicio y un fin. El inicio de una etiqueta es normal, en otras palabras, lleva los dos caracteres que mencioné anteriormente y dentro de estos se encuentra el nombre, sin embargo, existe una diferencia para el fin de la etiqueta ya que antes de escribir el nombre de la etiqueta debemos escribir una diagonal /. Como en este ejemplo:

<Inicio de la etiqueta>

</Fin de la etiqueta>

Este lenguaje fue desarrollado por la Organización Europea de Investigación Nuclear (CERN) en el año 1945 con la finalidad de desarrollar un sistema de almacenamiento donde las cosas no se perdieran, que pudieran ser conectadas a través de hipervínculos. Primeramente, crearon un dispositivo llamado «memex», el cual era considerado como un suplemento para la memoria.

Posteriormente, Douglas Engelbart, diseñó un entorno de trabajo por computadora que recibiría el nombre de oNLine System que poseía un catálogo para facilitar la tarea de búsqueda dentro de un mismo organismo.

Recién en 1965, Ted Nelson acuñó el término hipervínculo, ideando una estructura que se encontraba conectada de forma electrónica y que más tarde permitiría la creación de la World Wide Web (1989), un sistema de hipertexto a través del cual era posible compartir una variada información sirviéndose de Internet (servía para la comunicación entre investigadores nucleares que formaran parte del CERN).

El norteamericano Tim Berners-Lee fue el primero en proponer una descripción de HTML en un documento que publicó en 1991. Allí describía veintidós componentes que suponen el diseño más básico y simple del HTML.

El tipo de codificación que se utilizó para el desarrollo de este sistema de hipervínculos debía ser comprendido, tanto por ordenadores tontos como por mega-estaciones, por eso fue necesario crear uno absolutamente simples, tanto en lo que respectaba al lenguaje de intercambio (HTML), como el que hacía referencia al protocolo de red (HTTP).

Al día de hoy existen los Editores Web que permiten que los diseñadores, a través de herramientas gráficas que reciben el nombre de WYSIWYG puedan crear páginas web sin conocer el código html, este se crea de forma automatizada, dándole estructura a la web y permitiendo que sea más allá del ordenador donde es creada. Entre los recursos que pueden enlazarse al código HTML se encuentran fotografías, vídeos, archivos de otras webs o incluso de la misma y todo tipo de contenido que se encuentre subido a la red.



PHP

PHP, acrónimo recursivo en inglés de **PHP: Hypertext Preprocessor** (preprocesador de hipertexto), es un lenguaje de programación de propósito 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 un 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 el HTML resultante.

PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes. Puede ser usado en la mayoría de los servidores web al igual que en muchos sistemas operativos y plataformas sin ningún costo.

Fue creado originalmente por Rasmus Lerdorf en el año 1995. Actualmente el lenguaje sigue siendo desarrollado con nuevas funciones por el grupo PHP. Este lenguaje forma parte del software libre publicado bajo la licencia PHPv3_01, una licencia Open Source validada por Open Source Initiative. La licencia de PHP es del estilo de licencias BSD, sin la condición de *copyleft asociada con la* Licencia Pública General de GNU.



CSS

CSS (siglas en inglés de Cascading Style Sheets), en español "Hojas de estilo en cascada", es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Es muy usado para establecer el diseño visual de los documentos web, e interfaces de usuario escritas en HTML o XHTML; el lenguaje puede ser aplicado a cualquier documento XML, incluyendo XHTML, SVG, XUL, RSS, etcétera. También permite aplicar estilos no visuales, como las hojas de estilo auditivas.

Junto con HTML y JavaScript, CSS es una tecnología usada por muchos sitios web para crear páginas visualmente atractivas, interfaces de usuario para aplicaciones web y GUIs para muchas aplicaciones móviles (como Firefox OS).

CSS está diseñado principalmente para marcar la separación del contenido del documento y la forma de presentación de este, características tales como las capas o layouts, los colores y las fuentes. Esta separación busca mejorar la accesibilidad del documento, proveer más flexibilidad y control en la especificación de características presentacionales, permitir que varios documentos HTML compartan un mismo estilo usando una sola hoja de estilos separada en un archivo `.css`, y reducir la complejidad y la repetición de código en la estructura del documento.

La separación del formato y el contenido hace posible presentar el mismo documento marcado en diferentes estilos para diferentes métodos de renderizado, como en pantalla, en impresión, en voz (mediante un navegador de voz o un lector de pantalla, y dispositivos táctiles basados en el sistema Braille). También se puede mostrar una página web de manera diferente dependiendo del tamaño de la pantalla o tipo de dispositivo. Los lectores pueden especificar una hoja de estilos diferente, como una hoja de estilos CSS guardado en su computadora, para sobrescribir la hoja de estilos del diseñador.

La especificación CSS describe un esquema prioritario para determinar qué reglas de estilo se aplican si más de una regla coincide para un elemento en particular. Estas reglas son aplicadas con un sistema llamado de cascada, de modo que las prioridades son calculadas y asignadas a las reglas, así que los resultados son predecibles.

La especificación CSS es mantenida por el World Wide Web Consortium (W3C). El MIME type `text/css` está registrado para su uso por CSS descrito en el RFC 2318. El W3C proporciona una herramienta de validación de CSS gratuita para los documentos CSS.



MySQL

MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones. Sea cual sea el entorno en el que va a utilizar MySQL, es importante monitorizar de antemano el rendimiento para detectar y corregir errores tanto de SQL como de programación

Fue inicialmente desarrollado por MySQL AB (empresa fundada por David Axmark, Allan Larsson y Michael Widenius). MySQL A.B. fue adquirida por Sun Microsystems en 2008, y ésta a su vez fue comprada por Oracle Corporation en 2010, la cual ya era dueña desde 2005 de Innobase Oy, empresa finlandesa desarrolladora del motor InnoDB para MySQL.

Existen varias interfaces de programación de aplicaciones que permiten, a aplicaciones escritas en diversos lenguajes de programación, acceder a las bases de datos MySQL, incluyendo C, C++, C#, Pascal, Delphi (vía dbExpress), Eiffel, Smalltalk, Java (con una implementación nativa del driver de Java), Lisp, Perl, PHP, Python, Ruby, Gambas, REALbasic (Mac y Linux), (x)Harbour (Eagle1), FreeBASIC, y Tcl; cada uno de estos utiliza una interfaz de programación de aplicaciones específica.

Ventajas:

- MySQL software es Open Source
- 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.
- Su conectividad, velocidad, y seguridad hacen de MySQL Server altamente apropiado para acceder bases de datos en Internet
- 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).
- Un gran porcentaje de las utilidades de MySQL no están documentadas.
- No es intuitivo, como otros programas (ACCESS).



C#

Lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, que después fue aprobado como un estándar por la ECMA (ECMA-334) e ISO (ISO/IEC 23270).

C Sharp es uno de los lenguajes de programación diseñados para la infraestructura de lenguaje común.

.NET, las bibliotecas de clases fueron escritas originalmente usando un sistema de código gestionado llamado Simple Managed C (SMC). En abril de 1999, Anders Hejlsberg formó un equipo con la misión de desarrollar un nuevo lenguaje orientado a objetos.

En la actualidad existen los siguientes compiladores o IDE para el lenguaje C#:

Microsoft .NET Framework 2.0 (SDK) incluye un compilador de C#, pero no un IDE.

Microsoft Visual Studio, IDE por excelencia de este lenguaje.

SharpDevelop, IDE libre para C# bajo licencia GNU LGPL, con una interfaz muy similar a Microsoft Visual Studio.

Mono, es una implementación con licencia GNU GPL de todo el entorno .NET desarrollado por Novell. Como parte de esta implementación se incluye un compilador de C#.

DotGNU Portable.NET, de la Free Software Foundation.

Tipos de datos:

C# contiene veinte categorías generales de tipos de datos integrados: tipos de valor y tipos de referencia. El término tipo de valor indica que esos tipos contienen directamente sus valores. Tipos para definir números enteros:

- Tipo de datos de enteros:
byte, sbyte, short, ushort, int, uint, long, ulong.
- Tipo de datos de coma flotante:
float, double, decimal, monetario.
- Tipo de datos de caracteres:
char
- Tipo de datos lógicos:
bool

JAVASCRIPT

Es una especie de lenguaje de programación ligera, interpretado por la mayoría de los navegadores y que les proporciona a las páginas web, efectos y funciones complementarias a las consideradas como estándar HTML. Este tipo de lenguaje de programación, con frecuencia son empleados en los sitios web, para realizar acciones en el lado del cliente, estando centrado en el código fuente de la página web.

Fue creado por la compañía de software “Netscape Corporation” para que fuese colocado en su navegador 2.0 y que, gracias a su simplicidad, aún continúa siendo una de las herramientas de gran utilidad, para la creación de páginas web que posean algo más que texto.

Otras de las empresas que desarrollo Java Script fue Microsoft, para su Explorer, que por lo general es similar al Netscape.

Es importante que quede claro, que JavaScript no es del todo un lenguaje de programación, sino más bien un lenguaje de script (rutinas o guiones). Por lo tanto, es más parecido a los macros de los procesadores de hojas de cálculo o texto. Sería imposible ejecutar un programa completo con JavaScript.

Los Java Script ayudan a mejorar la gestión cliente/servidor; entre sus funciones básicas se encuentran: abrir y cerrar ventanas; cambios eficaces en una página (en lo que respecta a su contenido y aspecto; desarrollo de cadenas de texto; procedimientos aritméticos.

Dado que su misión es extender el HTML, Java Script es un lenguaje que contempla ciertas restricciones, que, de manera indirecta, terminan por brindarle seguridad al usuario.



SQL SERVER

Es un sistema de gestión de bases de datos relacionales (RDBMS) de Microsoft que está diseñado para el entorno empresarial. SQL Server se ejecuta en T-SQL (Transact -SQL), un conjunto de extensiones de programación de Sybase y Microsoft que añaden varias características a SQL estándar, incluyendo control de transacciones, excepción y manejo de errores, procesamiento fila, así como variables declaradas.

Bajo el nombre código Yukon en su etapa de desarrollo, SQL Server 2005 fue lanzado en noviembre de 2005. Se dice que el producto 2005 proporcionó una mayor flexibilidad, escalabilidad, confiabilidad y seguridad a las aplicaciones de base de datos, y permitió que fueran más fáciles de crear y desplegar, lo que reduce la complejidad y el tedio involucrado en la gestión de bases de datos. SQL Server 2005 también incluía más soporte administrativo.

El código original de SQL Server ha sido desarrollado por Sybase; a finales de 1980, Microsoft, Sybase y Ashton-Tate colaboraron para producir la primera versión del producto, SQL Server 4.2 para OS/2. Posteriormente, tanto Sybase como Microsoft ofrecieron productos de SQL Server. Sybase cambió después el nombre de su producto a Adaptive Server Enterprise