

Análisis Comparativo entre ASP.NET y PHP

Comparative Analysis between ASP.NET and PHP

Adriana Yolanda Sierra Cedeño
Analista Senior de Aplicaciones
Junta de Beneficencia de Guayaquil- ECUADOR
Estudiante Universidad Ecotec - ECUADOR

Marcos Antonio Espinoza Mina, Mgs.
Doctorando en Administración de Empresas
Universidad Católica Santa María de Buenos Aires-Argentina
Universidad Ecotec - ECUADOR
Universidad Agraria del Ecuador - ECUADOR
Autor corresponsal: yolita1175@hotmail.com, mespinoza@uagraria.edu.ec
Fecha de recepción: 26 de Septiembre de 2017 - Fecha de aceptación: 15 de Abril de 2018

Resumen: La tecnología avanza cada día con más velocidad, hace pocos años el uso del internet no era un asunto de primera necesidad como lo es ahora, si alguien indicaba que la internet desplazaría a la prensa escrita, hubiese sido ignorado, pero esa es la realidad actual, por lo cual la carrera hacia el uso de la web se ha vuelto muy competitiva y todas las empresas se están encaminando en esa dirección. Los desarrolladores informáticos se orientan al desarrollo web y es una necesidad conocer cuál lenguaje de programación es el idóneo para la elaboración de un sitio web. A través del método comparativo de investigación este trabajo se orientó hacia el reconocimiento y análisis de dos de las herramientas de desarrollo web más utilizadas en la actualidad como son ASP.NET y PHP. Se presentan los beneficios y falencias de cada una de ellas, exploradas por diversos autores, dándole al lector una idea general de lo que se puede alcanzar. Es un apoyo importante para la definición de la herramienta más adecuada para el desarrollador de aplicaciones web. Se consideran los aspectos más relevantes de ambas herramientas para dar mayor claridad al momento de decidir cuál escoger. Es importante conocer las opciones que brinda el lenguaje de programación por seleccionar ya sean estas funcionales o administrativas. Se concluye que a la hora de seleccionar el lenguaje de programación se debe de tener en cuenta todas las características y definiciones planteadas que harán del software un producto de calidad que agregue valor a la operación de la empresa.

Palabras clave: ASP.NET, PHP, herramientas, desarrollo, programación

Abstract: Technology is progressing more and more quickly, a few years ago the use of the internet was not a matter of necessity as it is now, if someone indicated that the Internet would displace the written press, it would have been ignored, but that is the current reality, so the race towards the use of the web has become very competitive and all the companies are heading in that direction. The computer developers are oriented to the web development and it is a necessity to know which programming language is the ideal one for the elaboration of a website. Through the comparative research method this work was oriented towards the recognition and analysis of two of the most commonly used web development tools such as ASP.NET and PHP. The benefits and shortcomings of each of them, explored by different authors, are presented, giving the reader a general idea of what can be achieved. It is an important support for the definition of the most appropriate tool for the web application developer. The most relevant aspects of both tools are considered to give greater clarity when deciding which to choose. It is important to know the options offered by the programming language to select whether these are functional or administrative. It is concluded that when selecting the programming language should take into account all the features and definitions that will make the software a quality product that adds value to the operation of the company.

Key words: ASP.NET, PHP, tools, development, programming

Introducción

En la actualidad la mayoría de los nuevos desarrollos de software están orientados hacia la web; tener un dispositivo de escritorio en el cual solo es posible acceder a un lugar específico ya no es una opción, las necesidades actuales exigen acceder a información de cualquier lugar y desde donde sea necesario. Es posible asegurar que las Tecnologías de la Información y las Comunicaciones (TIC) permiten desarrollar nuestros conocimientos e intercambiar criterios con otras personas sin importar la distancia, a través de las herramientas disponibles en Internet, como los cursos online, los foros de discusión entre muchas otras opciones, permitiendo de esta forma ampliar nuestro nivel cultural. Por tanto, todos los desarrollos de las nuevas aplicaciones en informática se han orientado hacia la web; hoy no existe algo que no se pueda hacer por esta vía, por este motivo es necesario conocer un poco más sobre las tecnologías que se están utilizando para desarrollar este tipo de aplicaciones (Díaz, Pérez, & Florido, 2011). La tecnología de implementación y las herramientas son factores muy importantes, y la elección de las mismas puede tener una influencia significativa en el éxito de un proyecto (Bulajic, Sambasivam, & Stojic, 2013). Los proyectos más costosos son mejor recibidos bajo el paradigma abierto y los programas libres poseen una tendencia natural a la competencia, mientras que el software propietario se comporta mejor en ambientes dominados por amplios márgenes de beneficio (Márquez, 2012).

Muchas veces se presentan soluciones que parecen hacer exactamente lo que se busca y siempre es tentador aceptarlas con los brazos abiertos, pero antes hay que consultar a la persona encargada de instalarlas en el entorno de trabajo. ¿Se basan en servidores Windows con SQL Server?, ¿se dispone de las licencias de uso?, ¿el soporte informático aceptaría el mantenimiento de esos servidores? (Pedraza & Banco, 2013).

Este trabajo tiene por objetivo analizar dos lenguajes de programación web para que sirva al lector como una orientación y pueda decidir cuál de los dos puede utilizar para dar solución a sus necesidades; adicionalmente se aportan ideas para que sea posible explotar más las herramientas.

Las herramientas a analizar son: ASP.Net y PHP, ambas permiten realizar páginas web dinámicas y son muy usadas en la actualidad, la selección de estos lenguajes se basa principalmente en el conocimiento previo de las mismas.

Para avanzar, es necesario establecer algunos conceptos y referencias básicas:

Lenguaje de Programación: Conjunto de reglas o normas que permiten asociar a cada programa correcto un cálculo que será llevado a cabo por un ordenador (sin ambigüedades), por tanto, un lenguaje de programación es un convenio o acuerdo acerca de cómo se debe de interpretar el significado de los programas de dicho lenguaje (Almagro, 2012).

Framework: Es una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación (Gutiérrez, 2014).

ASP.Net: Es una tecnología que permite generar y controlar páginas Web dinámicas con facilidad. Es la nueva generación de páginas Activas del Servidor (ASP) (Payne, 2002).

PHP: Nació como un lenguaje para hacer seguimiento de visitas a páginas personales, pero actualmente es uno de los lenguajes de scripts más populares (Cobo, 2005).

Aplicaciones Web: Son todas aquellas aplicaciones cuya interfaz se construye a través de páginas web (Berzal, Cortijo, & Cubero, 2004).

ASP.Net es un software comercial y PHP es libre. Desde el punto de vista técnico, a nivel general, se puede indicar que el software libre puede ser una buena solución tanto como lo es el software comercial. Desde el punto de vista del costo de implantación, -capacitación del personal, soporte técnico y disponibilidad- la competencia es realmente dura entre estas dos alternativas, aunque la mayoría de los casos se inclinan en favor del software comercial. En lo que respecta al desarrollo económico local y regional, puede ser atractivo adoptar una política pública que favorece la utilización de software libre porque impulsa la creación de empresas, pero esto puede ser arma de dos filos si la decisión no está sustentada estratégicamente (Menchaca García, 2004).

Estudios señalan que la reutilización de código es amplia y que los desarrolladores de software de código abierto, al igual que los desarrolladores de empresas, aplican herramientas que reducen sus costos de búsqueda de conocimiento y código, evalúan la calidad de los componentes de software y tienen incentivos para reutilizar código (Haefliger, von Krogh, & Spaeth, 2008).

El modelo de código abierto es en gran medida coherente con las actuales teorías económicas sobre las formas óptimas de aprovechar la Investigación y Desarrollo (I+D) para satisfacer las distintas necesidades de los diferentes mercados de usuarios finales. Existen algunas razones para creer que las empresas están siendo atraídas al desarrollo de código abierto como un segundo mejor resultado: ya que se hace cada vez más difícil mantener una diferenciación competitiva con una estructura de desarrollo tradicional, el código abierto ofrece una táctica alternativa prometedora (Mann, 2006).

A continuación se revisan las generalidades, ventajas, desventajas tanto de ASP.Net como de PHP, adicionalmente se presenta un cuadro comparativo de las principales diferencias entre ambos, mediante el método de comparación, el cual ha permitido que al finalizar el documento se pueda tener una clara idea de lo que cada lenguaje analizado es capaz de realizar y cuál es la diferencia entre el uso de uno u otro.

2. Desarrollo

A través de una revisión del contenido tecnológico documentado de varios autores, se presentan los puntos más sobresalientes de las generalidades, ventajas, desventajas de ASP.Net y de PHP, para posteriormente presentar un resumen comparativo de estas importantes herramientas de programación orientada a la web.

2.1. ASP.NET

2.1.1. Consideraciones Generales

ASP.NET, es la evolución de ASP que libera al programador de la utilización de extensas líneas de código que luego acarrearán en un seguimiento de errores igualmente extenso ya que el código embebido en HTML se vuelve complejo. Con las mejoras realizadas en ASP.NET el código se vuelve mucho más fácil de reutilizar y de depurar. En el desarrollo, se utilizan formularios web similares a los que se utilizan en Visual Basic (Berzal, Cortijo, & Cubero, 2004).

ASP.NET forma parte de la plataforma .NET (Figura 1), aunque no es posible intercambiar los formularios de Windows con los de los de ASP.NET. (Berzal, Cortijo, & Cubero, 2004)

Las páginas Web ASP.NET están completamente orientadas a objetos. En las páginas Web ASP.NET se puede trabajar con elementos HTML que usen propiedades, métodos y eventos. El marco de trabajo de páginas ASP.NET quita los detalles de implementación relacionados con la separación de cliente y servidor inherente a las aplicaciones web presentando un modelo unificado que responde a los eventos de los clientes en el código que se ejecuta en el servidor. El marco de trabajo también mantiene automáticamente el estado de la página y de los controles que contenga durante el ciclo vital de procesamiento de la página. (Microsoft, 2007)

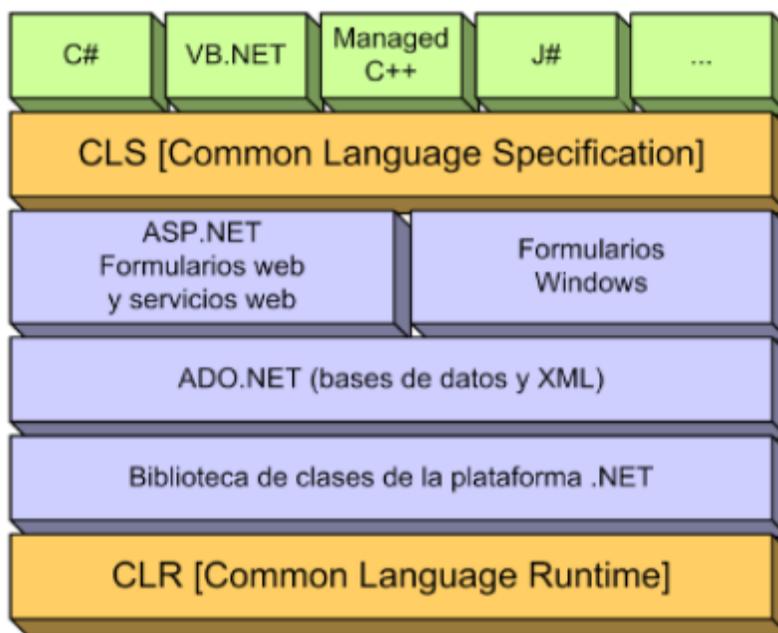


Figura 1: Plataforma ASP.Net Fuente: Desarrollo Profesional de Aplicaciones Web con ASP.NET

Según Padilla (2006) entre las principales características de ASP.Net se encuentran las siguientes:

- Eficiencia: se ha mejorado la tecnología para que desde el momento de la compilación se maneje eficientemente los recursos del sistema. (Padilla, 2006)

- Soporte de Lenguajes: es posible programar tanto en Visual Basic .net como en C# .net.
- Código y contenido por Separado: es mucho más sencilla la depuración ya que separa la interfaz de usuario con el código.
- Compatibilidad de Navegadores: se puede controlar de una mejor forma la navegación en varios navegadores.
- Código Compilado: en este entorno se realiza la compilación Just-in-time aumentando el rendimiento mediante el soporte nativo y servicios de caché.
- Controles de Servidor: permite al programador una herramienta para la creación de aplicaciones multiplataforma, y ahorrando muchas líneas de código.

Según Azofeifa (2011) una característica adicional es:

- Seguridad: Está vincula naturalmente con Microsoft Information Server (IIS), empleando autenticación Windows integrada (NTLM o Kerberos), o formularios mediante Microsoft Passport. (Azofeifa, 2011)

Según Guerra (2015) las siguientes son las principales características de ASP.Net:

- Programación Orientada a Objetos: La plataforma fue construida aplicando el paradigma de Programación Orientada a Objetos (POO). El núcleo de lenguajes como C# están basados en los principios OO.
- Soporte para múltiples lenguajes: En .NET, la verdadera interoperabilidad entre lenguajes es posible gracias a las capacidades que tiene la plataforma como herencia entre lenguajes (Cross-Language Interoperability) que junto con un sistema de tipos unificado (Common Type System), hace que la integración entre el código escrito en diferentes lenguajes sea total. Esto permite que se puedan usar otros paradigmas de programación tales como la programación funcional con F# o lenguajes dinámicos como Ruby o Python.
- Fácil desarrollo basado en componentes: En la plataforma .NET es más fácil implementar componentes o bibliotecas de componentes que comparten funcionalidades. La unidad de código compartido en .NET se denomina ensamblado (assembly), que lleva información de la versión y todos los metadatos necesarios para usarlo.
- Simplifica el despliegue de las aplicaciones: En contraste con las aplicaciones basadas en componentes COM, no es necesario el registro de los ensamblados, Con un “Xcopy Deployment” es suficiente, es decir con copiar los ensamblados es suficiente. Se ha eliminado por completo el clásico problema de DLL HELL, gracias a que múltiples versiones de un ensamblado pueden coexistir en la misma máquina. Un ejemplo de esto es el propio .NET Framework, que es posible tener diferentes versiones instaladas.
- Soporte para Biblioteca de Clases Base (Base Class Library): .NET Framework viene con un conjunto de bibliotecas de clases que proveen bloques básicos para construir aplicaciones, todas se proporcionan de manera consistente y están diseñadas bajo los principios de la POO. Ejemplos de estas bibliotecas incluyen el uso de colecciones, manipulación de texto, acceso a bases de datos, manipulación del sistema de archivos, etc.
- Implementación de varios tipos de aplicaciones: Gracias a la Biblioteca de Clases Base (BCL) es muy fácil el poder implementar cualquier tipo de aplicación. Ya sean basadas en escritorio (Windows Forms y Windows Presentation Foundation [WPF]), aplicaciones

Web (ASP.NET), servicios Web (Windows Communication Foundation [WCF]), aplicaciones para dispositivos inteligentes (Compact Framework), aplicaciones móviles para Windows Phone, aplicaciones Cloud (Microsoft Azure) y más.

- **Infraestructura de tiempo de ejecución unificada:** En el núcleo de .NET Framework se asienta el Common Language Runtime (CLR), que proporciona la infraestructura de tiempo de ejecución unificada en la forma de un lenguaje intermedio (IL) que comparten todos los lenguajes que se ejecutan en la plataforma, compilación Just-in-Time (JIT) de dicho código IL a código nativo para la máquina en la que se está ejecutando, gestión de memoria automática a través de un recolector de basura (Garbage Collector), etc.
- **Interoperabilidad con código existente:** La plataforma .NET tiene muy buen soporte para reutilizar componentes de software existentes que fueron escritos por tecnologías antiguas como los componentes COM y para proporcionar acceso a funciones del API de Win32 nativas del sistema operativo a través de un mecanismo llamado P/Invoke.
- **Manejo de excepciones:** El tratamiento de errores en el .NET Framework se proporciona a través de un mecanismo conocido como el Error Handling. Esto elimina las comprobaciones manuales que se hacían en la programación de Win32 y basada en COM con los llamados HRESULT.
- **Modelo de seguridad mejorada:** El motor en tiempo de ejecución de la plataforma .NET tiene un mecanismo de seguridad integrada llamado Code Access Security (CAS). Este modelo de seguridad es ortogonal a los mecanismos de seguridad proporcionados por el sistema operativo, como acceso control lista (ACL) y tokens de seguridad de Windows.
- **Soporte para comunicación con Servicios Web:** Desde el inicio de la plataforma .NET Framework, ha tenido soporte para servicios web como una forma de realizar llamadas a procedimientos remotos basadas en Simple Object Access Protocol (SOAP) y XML y poder establecer una Arquitectura Orientada a Servicios (SOA).

2.1.2. Ventajas

Según Padilla (2016), dentro de las muchas ventajas de ASP.NET se mencionaran las más relevantes:

- **Mejor Rendimiento:** Es capaz de aprovechar las ventajas del enlace anticipado, la compilación, la optimización nativa y los servicios de caché incrementan su rendimiento.
- **Compatibilidad con herramientas de primer nivel:** Su ambiente de trabajo es muy amigable como por ejemplo: utiliza controles de arrastrar y colocar.
- **Eficacia y Flexibilidad:** Todas las bibliotecas de clases, mensajería y las soluciones de acceso a datos se encuentran disponibles en la web y es posible utilizarlo con el lenguaje que mejor se adapte brinda al programador libertad para lo que necesite desarrollar.
- **Simplicidad:** Al mantener la lógica separada de la interface permite mayor facilidad para la implementación de servicios de código administrado como la recolección de elementos no utilizados (GarbageCollector).
- **Seguridad:** Con la configuración de la aplicación y la autenticación de Windows, es posible una seguridad completa para sus aplicaciones.(Figura 2)

.NET Framework 4 Security Architecture

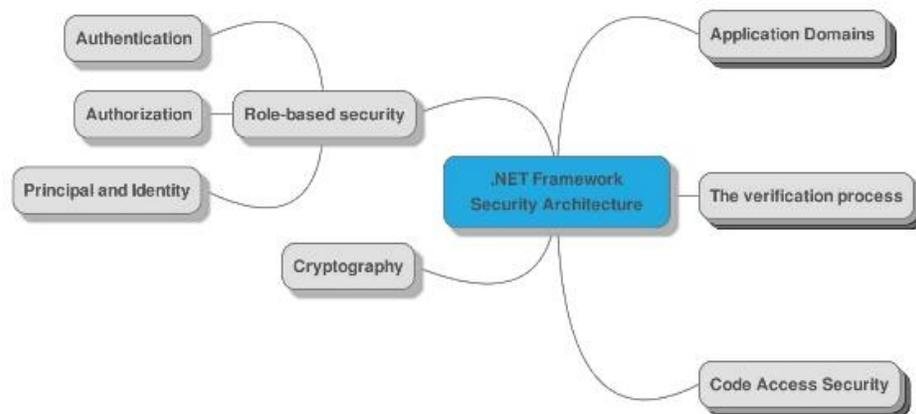


Figura 2: Arquitectura de Seguridad .Net Fuente: Mikhail Shcherbakov

Según Subgurin (2006), a continuación se enumera algunas otras ventajas de ASP.Net:

- Caché: se puede almacenar en la caché del servidor tanto páginas enteras, como controles personalizados o simples variables. En páginas críticas con mucha carga de base de datos nos es muy útil almacenar datos de la base de datos en la caché, reduciendo enormemente el consumo de recursos.
- Carpetas especializadas, como por ejemplo app_code que compila automáticamente las clases que se alojan en él, o la carpeta app_theme que alojan ficheros que marcan los temas de estilos de la Web.
- Los archivos de configuración Web.config y Machine.config permiten realizar operación de configuración en ficheros que hasta ahora había que realizar en el servidor.
- La adaptación automática del código devuelto a los dispositivos que le acceden. Una misma página puede servir para el Internet Explorer, para el Pocket Internet Explorer desde una PDA o para un navegador de un móvil cualquiera.
- La eliminación total de la necesidad de frames con la introducción de las masterpages.
- La extraordinaria compatibilidad con XML y los servicios Web.
- La multitud de controles Web que permiten mucha funcionalidad con poco código. Desde enlace con las bases de datos o enseñar fácilmente todos los datos, hasta simples etiquetas, hiperenlaces o generadores de imágenes.
- Se puede utilizar hasta cuarenta lenguajes distintos para el desarrollo en ASP.NET, aunque en el 95% de las aplicaciones se usa C#, VB.NET o J#.

2.1.3. Desventajas

Según Pérez (2015), es muy poco lo que se puede decir sobre desventajas de ASP.NET entre las cuales:

- Es un código propietario de Microsoft no abierto, se debe pagar por licenciamiento.

- Muchos hostings y alojamientos web no lo soportan por su alto coste. ASP.NET necesita tener instalado IIS con el Framework .Net, requiere una infraestructura robusta.
- El consumo de recursos de ASP.NET es importante por lo que se requieren servidores de mayor capacidad.

2.2. PHP

2.2.1. Consideraciones Generales

El lenguaje PHP (PHP Hypertext Pre-Processor) es uno de los más antiguos (fue creado en 1995 por la empresa PHP Group) y utilizado en el diseño de páginas web que utilizan bases de datos.

Se trata de un lenguaje interpretado en el lado del servidor que permite la creación de páginas web dinámicas que pueden estar dentro de páginas en HTML. Es uno de los lenguajes de programación web más populares por su rapidez y la facilidad de desarrollo. (Perez, 2015)

El código PHP se incluye entre etiquetas especiales de comienzo y final que permiten entrar y salir del modo PHP, es simple para el principiante, pero a su vez, ofrece muchas características avanzadas para los programadores profesionales. (Bakken, 1997)

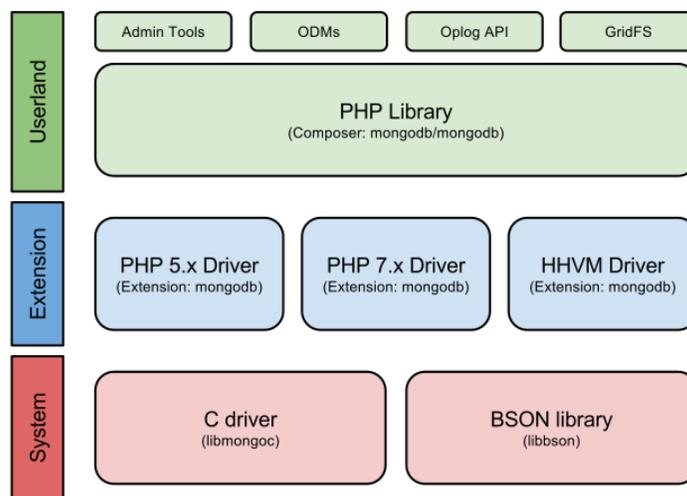


Figura 3: Arquitectura PHP Fuente: PHP

Según PHP (2017), La nueva arquitectura de PHP (Figura 3) está compuesta de la siguiente manera:

- En lo más alto se sitúa una biblioteca de PHP pura, la cual se distribuye como un paquete de Composer. Esta biblioteca proporcionará una API e implementa especificaciones comunes, para mejorar la consistencia de la API a través de todos los controladores mantenidos por MongoDB.
- Luego se encuentran los controladores de nivel más bajo: uno por plataforma. Estas extensiones formarán de forma efectiva la unión entre PHP, HHVM y las bibliotecas del

sistema (libmongoc y libbson). Estas extensiones expondrán una API pública idéntica para la funcionalidad más esencial y sensible al rendimiento:

- Administración de conexiones
- Codificación y decodificación de BSON
- Serialización documentos de objetos (para dar soporte a bibliotecas ODM)
- Ejecución de comandos y escritura de operaciones
- Manejo de consultas y cursores

Al separar el controlador y la API de alto nivel en extensiones y bibliotecas de PHP, respectivamente, se reduce la carga de mantenimiento y permitir una iteración más rápida en nuevas características. Además, una API pública idéntica para estas extensiones hará mucho más fácil portar una aplicación durante tiempos de ejecución de PHP, si la aplicación usa directamente el controlador de bajo nivel o una biblioteca de alto nivel.

GridFS (Utilidad para almacenar y extraer ficheros de la base de datos.) es una especificación de alto nivel. Su API es una abstracción para acceder a dos colecciones: ficheros (esto es, metadatos) y trozos (esto es, bloques de datos). Asimismo, todo el añadido sintáctico encontrado en el controlador antiguo de mongo, como el procesamiento de ficheros subidos o la exposición de ficheros de GridFS como flujos de PHP, se puede implementar en PHP puro. Siempre que se cuente con métodos rendidores para leer y escribir en colecciones de GridFS.

Algunos usuarios podrían preferir adherirse a la API sin florituras ofrecida por las extensiones, o crear su propia abstracción de alto nivel. Las futuras bibliotecas podrían incluir una biblioteca de PHP dirigida a la administración de MongoDB con una API para la gestión de varios usuarios y comandos de operaciones.

Según Bakken (1997), PHP puede hacer cualquier cosa que se pueda hacer con un script CGI, como procesar la información de formularios, generar páginas con contenidos dinámicos, o mandar y recibir cookies. Y esto no es todo, se puede hacer mucho más. Existen tres campos en los que scripts escritos en PHP son usados:

- Scripts en la parte del servidor. Este es el campo más tradicional y el principal campo de trabajo. Se necesitan tres cosas para que esto funcione. El procesador PHP (CGI o módulo), un servidor web y un navegador. Se necesita correr el servidor web con PHP instalado. El resultado del programa PHP se puede obtener a través del navegador, conectando con el servidor web. (Figura 4)
- Scripts en línea de comandos. Es posible crear un script PHP y correrlo sin ningún servidor web o navegador. Sólo es necesario el procesador PHP para usarlo de esta manera. Este tipo de uso es ideal para scripts ejecutados regularmente desde cron (en *nix o Linux) o el Planificador de tareas (en Windows). Estos scripts también pueden ser usados para tareas simples de procesamiento de texto.
- Escribir aplicaciones gráficas. PHP no es probablemente el mejor lenguaje para escribir aplicaciones gráficas, pero si se conoce bien PHP, y se utilizan algunas características avanzadas en programas clientes, es posible utilizar PHP-GTK *para escribir dichos programas. Es también posible escribir aplicaciones independientes de una plataforma. PHP-GTK es una extensión de PHP, no disponible en la distribución principal.

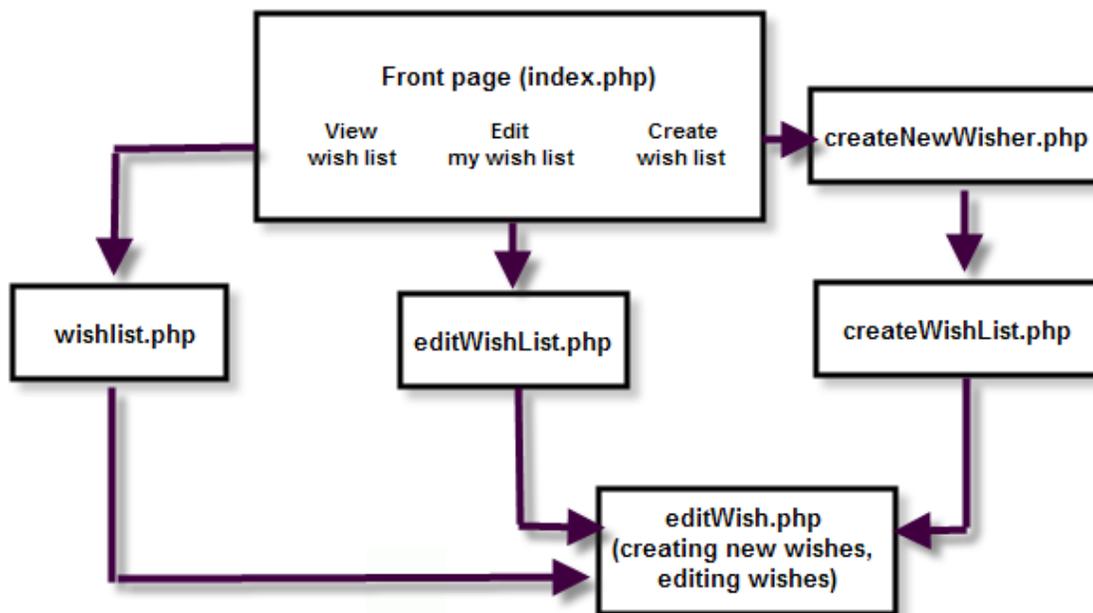


Figura 4 Estructura de sitio web en PHP Fuente: Web Engineering

Lo mejor de utilizar PHP es su extrema simplicidad para el principiante, pero a su vez ofrece muchas características avanzadas para los programadores profesionales. Una de las características más potentes y destacables de PHP es su soporte para un amplio abanico de bases de datos. Escribir una página web con acceso a una base de datos es increíblemente simple utilizando una de las extensiones específicas de bases de datos (p.ej., para mysql), o utilizar una capa de abstracción como PDO, o conectarse a cualquier base de datos que admita el estándar de Conexión Abierta a Bases de Datos por medio de la extensión ODBC. Otras bases de datos podrían utilizar cURL o sockets, como lo hace CouchDB. (My PHP.net, 2017)(Figura 5)

Con PHP no se está limitado a generar HTML. Entre las capacidades de PHP se incluyen la creación de imágenes, ficheros PDF e incluso películas Flash (usando libswf y Ming) generadas sobre la marcha. También se puede generar fácilmente cualquier tipo de texto, como XHTML y cualquier otro tipo de fichero XML. PHP puede autogenerar estos ficheros y guardarlos en el sistema de ficheros en vez de imprimirlos en pantalla, creando una caché en el lado del servidor para contenido dinámico (My PHP.net, 2017).

PHP también cuenta con soporte para comunicarse con otros servicios usando protocolos tales como LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (en Windows) y muchos otros. También se pueden crear sockets de red puros e interactuar usando cualquier otro protocolo. PHP tiene soporte para el intercambio de datos complejos de WDDX entre virtualmente todos los lenguajes de programación web. Y hablando de interconexión, PHP tiene soporte para la instalación de objetos de Java y emplearlos de forma transparente como objetos de PHP. (SitioOficial)

PHP puede combinarse con C para lograr un mejor manejo de cálculos matemáticos, para que esta fusión sea exitosa el desarrollador debe conocer el uso de los elementos internos de PHP, así se optimizará el uso de la memoria. (Šimec, Lozić, & Tepeš Golubić, 2017)

Es posible adaptar UDFs a matrices esparcidas estructuradas, tales como matrices diagonales, escasamente simétricas matrices triangulares, matrices de Toeplitz, etc. Es posible identificar otras operaciones matemáticas aplicables que se pueden implementar con el entorno PHP / MySQL, mejorando así la funcionalidad de minería de datos del DBMS. Además, la aplicación matriz se basará en los principios de la programación orientada a objetos. (Tasić, Stanimirović, & Pepić, 2010)

PHP con MySQL, se ha utilizado en aplicaciones para la salud, dando resultados óptimos de rapidez y eficacia, incluso con aplicaciones móviles, garantizando los accesos seguros y la confidencialidad del paciente. (De la Torre-Díez, Antón-Rodríguez, Díaz-Pernas, & Perozo-Rondón, 2012)

2.2.2. Ventajas

Según PHP INFO-IUTEPI (2017), dentro de las ventajas de PHP se mencionaran las siguientes:

- Es un lenguaje sencillo y fácil de estudiar y aprender.
- Una de sus características es la rapidez.
- Lo soportan la mayoría de las plataformas de alojamiento web.
- Tiene ciertas características de los lenguajes orientados a objetos como la utilización de clases y herencias.
- Puede mezclarse con código HTML, aunque esto dificulta su lectura.
- Puede manejar ficheros y conectarse a distintas bases de datos (MySQL, Oracle, SQL Server, Informix, PostgreSQL, etcétera).
- El software que permite soportarlo en los servidores de hosting es libre y gratuito.
- Está en continuo desarrollo y soporta numerosas funcionalidades.
- Existe numerosa documentación sobre el lenguaje en Internet por lo que es relativamente sencillo resolver los problemas que nos puedan surgir durante el desarrollo de un sitio web.
- No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.(Figura 6)

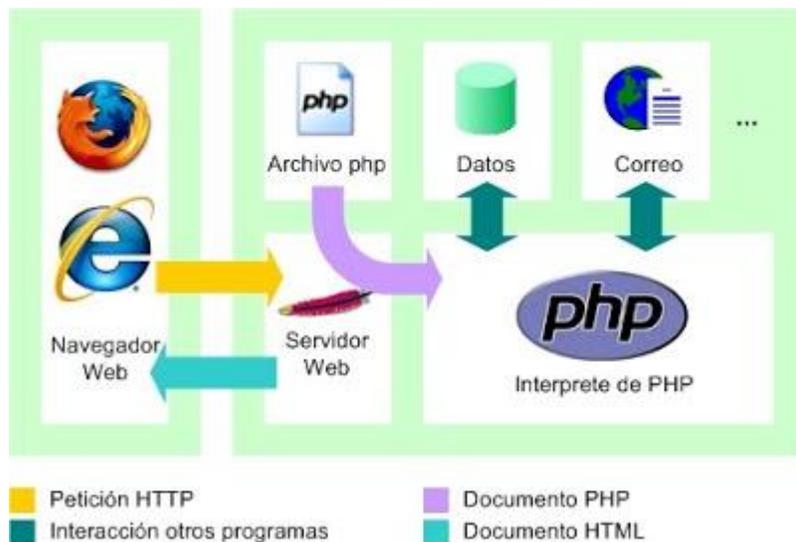


Figura 5: Ejecución de Página web PHP. Fuente: 3w.windows-az.com

Adicionalmente Aula Formativa (2014) indica que:

- El lenguaje PHP permite incluir caracteres Unicode a los códigos. Esto implica que los caracteres de muchos idiomas, alrededor del mundo pueden ser incluidos cuando se programa en PHP.
- Solo es necesario colocar el código HTML, CSS o de cualquier otro lenguaje listado en un archivo directamente. PHP se integra con ellos sin problemas.
- Este lenguaje contiene un manual completo online, que es accesible a cualquier programador deseoso de aprender. De modo, que sólo es necesario revisarlo para resolver dudas.
- Las aplicaciones desarrolladas en PHP serán compatibles en diversos sistemas operativos ya que éstas corren independiente en un servidor.
- Con el lenguaje PHP es posible crear sitios que se cargan más rápidamente pues trabaja desde su propio espacio de memoria, al contrario de otros lenguajes.
- PHP es compatible con diversas bases de datos, aunque la más empleada para estos casos sea MySQL. Pero para el lenguaje PHP no hay límites, puede conectarse con otras bases de datos.
- Para funcionar, PHP solo requiere un servidor de Linux y se puede acceder a esta característica sin ningún costo adicional por parte del servicio de hosting.
- Tal vez una de las causas de la creciente popularidad de PHP a que ayuda a los desarrolladores a implementar elementos seguros en sus aplicaciones. Para los hackers es complicado romper la seguridad de aplicaciones en PHP, lo que implica que es menos probable que existan violaciones de seguridad en las aplicaciones.

2.2.3. Desventajas

Entre las desventajas de PHP se pueden mencionar las siguientes:

- PHP no es probablemente el mejor lenguaje para escribir aplicaciones gráficas, pero si es posible utilizando PHP-GTK para escribir dichos programas. Es también posible

escribir aplicaciones independientes de una plataforma. PHP-GTK es una extensión de PHP, no disponible en la distribución principal. (Bakken, 1997)

- Para poder ver y testear las páginas que vayamos creando es necesario disponer de un servidor web que soporte PHP. (Perez, 2015)
- Parte del contenido de las páginas puede no ser accesible a los navegadores, dificultando el posicionamiento de las páginas. (Perez, 2015)
- Como es un lenguaje que se interpreta en ejecución, para ciertos usos puede resultar un inconveniente que el código fuente no pueda ser ocultado. La ofuscación es una técnica que puede dificultar la lectura del código pero no necesariamente impide que el código sea examinado (PHP INFO-IUTEPI, 2017).
- Debido a que es un lenguaje interpretado, un script en PHP suele funcionar considerablemente más lento que su equivalente en un lenguaje de bajo nivel, sin embargo este inconveniente se puede minimizar con técnicas de caché tanto en archivos como en memoria (PHP INFO-IUTEPI, 2017).
- En las versiones previas a la 7, las variables no son tipificadas, lo cual dificulta a los diferentes IDEs ofrecer asistencias para el tipificado del código, aunque esto no es realmente un inconveniente del lenguaje en sí. Esto es solventado por algunos IDEs añadiendo un comentario con el tipo a la declaración de la variable. (PHP INFO-IUTEPI, 2017)

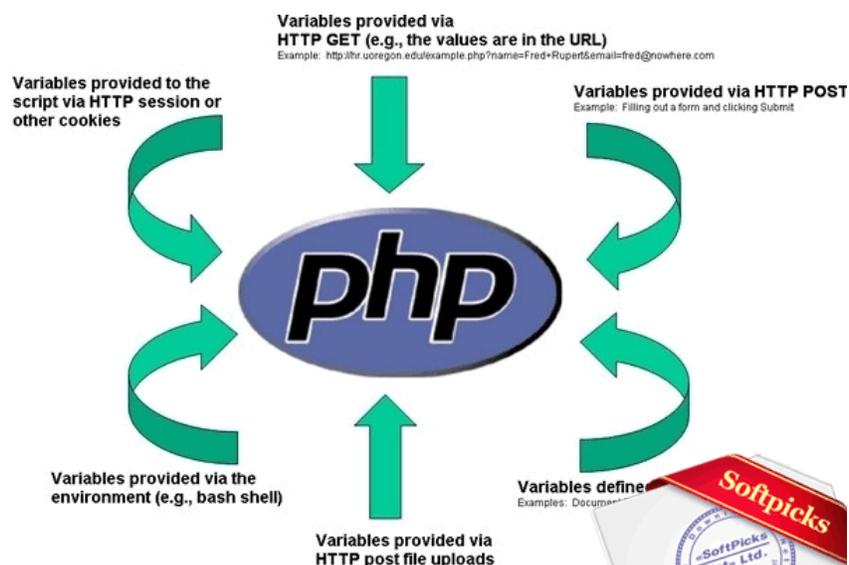


Figura 6 Utilización de Variables en PHP Fuente: PHP INFO-IUTEPI

2.3. Cuadro Comparativo ASP.NET vs PHP

Es evidente que tanto las licencias de uso como las licencias de código abierto cuentan con sus fortalezas y debilidades; la elección de unas u otras depende realmente de la naturaleza del software y el destino que el licenciante tenga para éste (Alvarado García, 2013).

El acceso al código fuente constituye un principio a partir del cual la comunidad puede continuar construyéndose a sí misma (Alonso & D'Antonio, 2015). Por el lado económico se debe añadir, que aunque parezca una gran ventaja la opción del software libre sobre el comercial,

no se establece que los productos de código abierto estén siempre libres de costes o que quienes producen software de código abierto no puedan vender y obtener ganancias financieras de los productos que desarrollan (Moreno & Cardoso, 2012). Adicionalmente desde la perspectiva legal, los riesgos asociados con el uso de software de código abierto, no son mínimos, pero son generalmente conocidos y un conjunto de herramientas eficaz para prevenir la mezcla de código abierto con código cerrado está disponible. Si la violación ocurre sin embargo, hay medidas que una empresa podría tomar para remediar la violación o para cumplir con los requisitos de licencia (Tsotsorin, 2013).

ASP.NET, a pesar de ser un producto licenciado, en el ambiente web también le da libertad, proporciona un control total sobre HTML y ofrece una interacción más limpia con JavaScript en línea. Esta libertad permite el desarrollo de soluciones puras de Ajax sin atar un sitio a un framework comercial específico (Esposito, 2011).

Regresando a las herramientas evaluadas, en base a los elementos técnicos expuestos anteriormente por los autores referenciados se construyó la siguiente tabla (Tabla1) que muestra las similitudes y diferencias entre ASP.Net y PHP.

Factores	ASP.Net	PHP
Código libre	No	Si
Rapidez en la ejecución	Regular	Si
Sintaxis del lenguaje	Si	Regular
Flexibilidad	Si	No
Seguridad	Si	Si
Desarrollo rápido de aplicaciones	Si	No
Soporta múltiples lenguajes	Si	No
Lenguaje orientado a objetos	Si	Regular
Soporte base de datos	Regular	Si
Control de datos	Si	No
Costo	Si	No
Soporte y documentación	Regular	Si
Manejo de excepciones	Si	Si
Soporte a aplicaciones móviles	Si	Regular
Recolección de elementos no utilizados	Si	Regular
Multihilo	Si	No

Tabla 1: Cuadro comparativo ASP.Net y PHP

Conclusiones

Es una tarea difícil comparar a estos dos lenguajes de programación, ASP.Net y PHP, debido a que son muy diferentes y además ambos son muy utilizados; pero como resultado de la investigación se puede indicar que ASP.Net tiene muchas más características útiles para el desarrollo.

Actualmente los desarrolladores se inclinan más hacia los desarrollos con herramientas de código abierto, sin embargo también existen aquellos que confían en el código propietario, cada uno con sus razones ya sean estas técnicas o no. En el mercado hay demanda para ambas corrientes.

El software que permite soportar a PHP en los servidores es libre y gratuito, lo que podría ser determinante por costos para empresas con poco presupuesto; sin embargo otras muchas empresas cuentan con infraestructura basada totalmente en Microsoft, lo cual hace que los desarrollos con herramientas .Net no tengan un costo adicional, en este punto se eliminaría la desventaja de ASP.Net en lo que respecta al costo. La decisión finalmente la toma el desarrollador de cual lenguaje utilizar.

La gran cantidad de información sobre PHP que existe en la web facilita su aprendizaje, situación que es mucho más difícil de lograr con ASP.Net el cual es más difícil de aprender, pero su entorno de desarrollo es mucho más amigable.

Tanto PHP con ASP.Net tienen buenos niveles de seguridad los cuales hacen las aplicaciones desarrolladas mucho más seguras e impenetrables a los hackers, dando un plus al código sobre otros lenguajes, una vez más la elección del lenguaje dependerá exclusivamente de la preferencia del desarrollador.

Cuando el desarrollador comienza aprendiendo ASP.Net y luego PHP, es común que le cueste mucho más trabajo utilizar PHP, debido a que en PHP no se cuenta con las funciones que ASP.Net proporciona para facilitar la labor, en PHP es necesario programar la mayoría de las funciones requeridas. Cuando sucede lo contrario, es decir comienza con PHP y luego conoce ASP.Net, el desarrollador se adapta pronto ya que en ASP.Net tendrá muchas facilidades.

La experiencia con ambos lenguajes es lo que determinará cuál es el mejor según cada desarrollador, este profesional es el único que decide que herramienta usar en su empresa.

Bibliografía

Šimec, A., Lozić, D., & Tepeš Golubić, L. (2017). *Benchmarking PHP modules*. Zagreb: Zagreb University of Applied Sciences.

3w.windows-az.com. (2011). Retrieved 2017 from <http://windows-az.com/full-php-vj-mysql-ilj-veb-proqramlaeddrma-vj-e.html>

- Aula Formativa. (5 de Agosto de 2014). *Aula Formativa*. Retrieved 2017 from <http://blog.aulaformativa.com/lenguaje-php-el-mejor-para-desarrollo-de-aplicaciones-web/>
- Azofeifa, D. M. (2011). *Programación Avanzada en Web*. UNED.
- Alvarado García, M. M. (2013). Análisis comparativo de los modelos de licenciamiento no personalizado de software desde el régimen de derecho de autor en el ordenamiento colombiano. *Revista de Derecho, Comunicaciones y Nuevas Tecnologías* (9).
- Almagro, C. U. (2012). *CCM*. Retrieved 2017 from Lenguajes de Programación: <http://es.ccm.net/contents/304-lenguajes-de-programacion>
- Alonso, A., & D'Antonio, S. (2015). El software libre y el Open Knowledge como comunidades de conocimiento paradigmáticas. *UTOPIA Y PRAXIS LATINOAMERICANA* (69), 83-92.
- Bulajic, A., Sambasivam, S., & Stojic, R. (2013). An Effective Development Environment Setup for System and Application Software. *Issues in Informing Science and Information Technology* , 10, 37-66.
- Bakken, S. S. (1997). *Manual de PHP*. Rafael Martínez.
- Berzal, F., Cortijo, F. J., & Cubero, J. C. (2004). *Desarrollo Profesional de Aplicaciones Web con ASP.NET*. iKor Consulting.
- Cobo, Á. (2005). *PHP y MySQL: Tecnología para el desarrollo de aplicaciones web*. Ediciones Díaz de Santos.
- Esposito, D. (2011). Which ASP.NET Is Better? *InformationWeek* , 15.
- De la Torre-Díez, I., Antón-Rodríguez, M., Díaz-Pernas, F. J., & Perozo-Rondón, F. J. (2012). *Comparison of Response Times of a Mobile-Web EHRs System Using PHP and JSP Languages*. Springer Science+Business Media.
- Díaz, J., Pérez, A., & Florido, R. (2011). Impacto de las Tecnologías de la Información y las Comunicaciones (TIC) para disminuir la brecha digital en la sociedad actual. *Cultivos Tropicales* , 5-10.
- Guerra, S. P. (08 de Junio de 2015). *System.OutOfMemoryException*. From <https://sparraguerra.wordpress.com/2015/06/08/net-caracteristicas-clave-y-ventajas-de-la-plataforma-net-framework/>
- Gutierrez, J. (2014). *¿Qué es un framework web?*. From http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework .
- Haefliger, S., von Krogh, G., & Spaeth, S. (2008). Code Reuse in Open Source Software. *Management Science* , 54 (1), 180 - 193.
- My PHP.net. (Septiembre de 2017). *My PHP.net*. From My PHP.net: <http://php.net/>
- Mann, R. J. (2006). Commercializing Open Source Software: Do Property Right Still Matter. *Harvard Journal of Law & Technology* , 20 (1), 1-47.
- Márquez, G. (2012). El mercado del software: código abierto y efectos de red. *Outlier, Economía* (6).
- Menchaca García, F. (08 de 2004). *¿Software libre o software comercial?* (P. d. línea, Ed.) From Política digital : <http://www.politicadigital.cl/?P=leernoticia&Article=598&c=114>

- Microsot. (2007). *Microsoft Developer Network*. From [https://msdn.microsoft.com/es-es/library/4w3ex9c2\(v=vs.100\).aspx](https://msdn.microsoft.com/es-es/library/4w3ex9c2(v=vs.100).aspx)
- Moreno, V., & Cardoso, J. (2012). Benefits and success factors of open-source web services development plataforms for small software houses. *JISTEM - Journal of Information Systems and Technology Management* , 9 (3), 585-606.
- Payne, C. (2002). *Aprendiendo ASP. net en 21 lecciones avanzadas*. Pearson Educación.
- Padilla, M. B. (2006). *ASP .Net orientado al desarrollo de aplicaciones web*. México.
- Pedraza, R., & Banco, S. C. (2013). Commercializing Open Source Software: Do Property Right Still Matter. *El profesional de la información* , 22 (1), 74-79.
- Perez, J. L. (2015). *Registro y registradores de dominios*. From <http://www.registrodominiosinternet.es/2013/08/lenguajes-programacion-web-ventajas.html>
- PHP INFO-IUTEPI. (2017). Retrieved 2017 from <https://sites.google.com/site/phpinfoiutepi/caracteristicas-de-php-1>
- SubGurim. (2006). Retrieved 2017 from <http://www.subgurim.net/Articulos/asp-net-general/3/por-que-asp-net.aspx>
- Samame, R. (2017). *BSGrupo*. From <https://bsgrupo.com/bs-campus/blog/ASP-vs-PHP-Cul-escojer-Ventajas-y-Desventajas-1127>
- SitioOficial. (n.d.). *PHP. Sitio Ofical*. From <http://php.net/manual/es/intro-whatcando.php>
- Tasiéc, M. B., Stanimirović, P. S., & Pepić, S. H. (2010). *Computation of generalized inverses using PHP/MySQL environment*. Serbia: Faculty of Science and Mathematics.
- Tsotsorin, M. V. (2013). Open Source Software Compliance: The Devil is Not So Black As He is Painted. *Santa Clara High Technology Law Journal* , 29 (3), 559-602.

Glosario

Acceso control lista (ACL). En inglés access control list es un concepto de seguridad informática usado para fomentar la separación de privilegios. Es una forma de determinar los permisos de acceso apropiados a un determinado objeto, dependiendo de ciertos aspectos del proceso que hace el pedido.

API. La abreviatura viene del término en inglés Application Programming Interfaces (Interfaces de programación de aplicaciones) son un conjunto de comandos, funciones y protocolos informáticos que permiten a los desarrolladores crear programas específicos para ciertos sistemas operativos.

Aplicaciones Cloud. Este tipo de servicios se ejecutan bajo un sistema diseñado para escalar bajo demanda, accesible mediante diferentes herramientas y soportar múltiples conexiones. Servicios como Dropbox o Gmail son considerados como cloud apps.

app_code. Es un directorio donde se guarda todos los archivos de código fuente.

app_theme. Se utiliza para guardar los temas del sitio.

Arquitectura Orientada a Servicios (SOA). En inglés Service Oriented Architecture) es un paradigma de arquitectura para diseñar y desarrollar sistemas distribuidos.

Biblioteca de Clases Base (BCL). Es una librería incluida en el .NET Framework formada por cientos de tipos de datos que permiten acceder a los servicios ofrecidos por el CLR y a las funcionalidades más frecuentemente usadas a la hora de escribir programas.

C# .net. Lenguaje de programación que combina los mejores elementos de múltiples lenguajes de amplia difusión como C++, Java, Visual Basic o Delphi. De hecho, su creador Anders Heljsberg fue también el creador de muchos otros lenguajes

CGI. Interfaz de entrada común (en inglés Common Gateway Interface, abreviado CGI) es una importante tecnología de la World Wide Web que permite a un cliente (navegador web) solicitar datos de un programa ejecutado en un servidor web. CGI especifica un estándar para transferir datos entre el cliente y el programa.

Code Access Security (CAS). Code Access Security (CAS), en el framework de Microsoft .NET, es la solución para evitar que el código no confiable realice acciones privilegiadas. Cuando el CLR carga un ensamblaje obtendrá evidencia para el ensamblaje y usará esto para identificar el grupo de códigos al que pertenece el ensamblaje.

Common Language Runtime (CLR). Es el encargado de compilar una forma de código intermedio llamada Common Intermediate Language (CIL, anteriormente conocido como MSIL, por Microsoft Intermediate Language), al código de maquina nativo, mediante un compilador en tiempo de ejecución.

CouchDB. Apache CouchDB. Comúnmente llamada CouchDB, es un gestor de bases de datos de código abierto, cuyo foco está puesto en la facilidad de su uso y en ser "una base de datos que asume la web de manera completa".

cURL. Es una librería de funciones para conectar con servidores para trabajar con ellos. El trabajo se realiza con formato URL. Es decir, sirve para realizar acciones sobre archivos que hay en URLs de Internet, soportando los protocolos más comunes

F#. Es un lenguaje de programación de código abierto desarrollado por Microsoft en base al lenguaje ML que conjuga la programación funcional con el desarrollo orientado a objetos.

HHVM. Es una máquina virtual para ejecutar PHP. Y entre otros lenguajes de programación.

IDE. Un entorno de desarrollo integrado o entorno de desarrollo interactivo, en inglés **Integrated Development Environment (IDE).** Es una aplicación informática que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de software.

J#. es un lenguaje transicional para programadores del lenguaje de programación Java y del lenguaje J++ de Microsoft, creado con la intención de que ambos puedan usar sus conocimientos actuales para crear aplicaciones en la plataforma .NET de Microsoft.

Kerberos. Es un protocolo de autenticación de redes de ordenador creado por el MIT que permite a dos ordenadores en una red insegura demostrar su identidad mutuamente de manera segura.

LDAP. Son las siglas de Lightweight Directory Access Protocol (en español Protocolo Ligero/Simplificado de Acceso a Directorios) que hacen referencia a un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

Lenguaje intermedio (IL). Es el lenguaje de una máquina abstracta diseñada para ayudar a realizar el análisis de un programa informático.

Libbson. Es una biblioteca que proporciona rutinas útiles relacionadas con la construcción, el análisis y la iteración de documentos BSON.

Libmongoc. Es la biblioteca de cliente oficial para aplicaciones C y proporciona una base para controladores MongoDB en lenguajes de nivel superior.

Libswf. Módulo de PHP.

Ming. Es una biblioteca de código abierto que le permite crear películas en formato Flash. Soporta casi todas las funciones de Flash 4, incluyendo: formas, degradados, mapas de bits, morphs, texto, botones, acciones, clips, streaming mp3 y transformaciones de color - lo único que falta son los eventos sonoros.

MongoDB. Esta extensión da soporte a PHP y HHVM, proporciona una API mínima para la funcionalidad principal del controlador: comandos, consultas, escrituras, gestión de conexiones y serialización BSON.