



TECNOLÓGICO  
NACIONAL DE MÉXICO



## CUADRO COMPARATIVO

### EQUIPO 6

#### PARTICIPANTES

CAÑEDO NÚÑEZ SEBASTIÁN ARATH  
DE HARO DE LA CRUZ ANA SOFÍA  
PASOS TEPOSTE RICARDO ALAN

#### PROFESORA

M.C. MARTHA PATRICIA SEVILLA ZAZUETA

#	<i>Criterios</i>	<i>Frameworks</i>	<i>Plug-ins</i>	<i>Componentes</i>
1	<b>Definición</b>	Es un entorno o ambiente de trabajo para desarrollo.	Son pequeños programas complementarios que amplían las funciones de aplicaciones web y programas de escritorio.	Un elemento de un sistema software que ofrece un conjunto de servicios, o funcionalidades, a través de interfaces definidas.
2	<b>Características</b>	Incluye bloques de construcción básicos de un programa. Determina el diseño de la estructura básica del software.	Funciona independiente del programa.	<ul style="list-style-type: none"> <li>❖ Estandarizado</li> <li>❖ Independiente</li> <li>❖ Accesible a ser compuesto</li> <li>❖ Desplegable</li> <li>❖ Documentado</li> </ul>
3	<b>¿Para qué se usa?</b>	Un Framework sirve para poder escribir código o desarrollar una aplicación de manera más sencilla.	Aportar funciones adicionales a aplicaciones web o programas informáticos.	Los componentes de software proporcionan las funciones en sus patrones de sistemas virtuales.
4	<b>¿Qué tipos existen?</b>	Para aplicaciones web, aplicaciones en general, tecnología AJAX, gestión de contenidos, de multimedia.	Existe una variedad casi infinita de plug-ins, pero algunos pueden ser los de seguridad, medición, compresión, de audio y video, sociales, etc.	<ul style="list-style-type: none"> <li>❖ Componentes de despliegue o distribución</li> <li>❖ Componentes de Producto de Trabajo</li> <li>❖ Componentes de Ejecución</li> </ul>
5	<b>Funcionalidad</b>	Simplifica la elaboración de una tarea, ya que solo es necesario complementarlo de acuerdo a lo que se quiere realizar.	Puede funcionar sin el programa principal real.	Un componente de software puede ser desplegado de forma independiente y puede participar en composiciones de terceras partes.

6	<b>¿Dónde son usados?</b>	Usualmente en escritorio para poder crear código.	Los plugins prácticamente son usados en todo ámbito de software y tipo de programa. A final de cuenta son complementos extra de un programa.	Ingeniería de software basada en componentes.
7	<b>Costo</b>	La gran mayoría de Frameworks no tienen costo de empleo	Existen Plugins gratuitos y Premium, los Premium pueden ir desde los \$10 USD hasta los \$400 USD.	Hay componentes comerciales en los que se debe pagar licencia, y otros que son gratis y existen bajo la premisa de Freeware y GNU.
8	<b>Ventajas</b>	<ul style="list-style-type: none"> <li>❖ Estructura y organización del código predeterminada.</li> <li>❖ Reutilización del código.</li> <li>❖ Agilidad y rapidez en el desarrollo.</li> <li>❖ Menor coste de desarrollo.</li> <li>❖ Buenas prácticas de desarrollo con el uso de patrones.</li> <li>❖ Minimizar errores y mayor facilidad para solucionarlos.</li> <li>❖ Facilita el mantenimiento.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Añaden funcionalidades adicionales.</li> <li>❖ Ahorra tiempo de trabajo en líneas de código.</li> <li>❖ Se puede configurar como mejor convenga.</li> <li>❖ Los administradores básicos pueden desarrollar esta tarea porque son sencillos de utilizar.</li> <li>❖ Tienen un panel de control para poder personalizarlos.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Puede ser mucho más barato comprar un componente comercial, donde el costo de desarrollo es amortizado por muchos clientes, que intentar desarrollar un nuevo software.</li> <li>❖ El componente comercial puede ser aceptado como un mejor producto al ser probado y validado por el vendedor y otros usuarios del componente en el mercado que uno construido por nosotros mismos.</li> <li>❖ Permite integrar nuevas tecnologías y nuevos estándares más fácilmente y rápidamente que si se construye por la propia organización.</li> </ul>

9	<b>Desventajas</b>	<ul style="list-style-type: none"> <li>❖ Tiempo de aprendizaje.</li> <li>❖ Versiones inestables.</li> <li>❖ Menor rendimiento.</li> <li>❖ Código sin utilizar.</li> <li>❖ Elección de Framework.</li> </ul>	<ul style="list-style-type: none"> <li>❖ El uso excesivo de ellos cuando no se necesitan puede ralentizar páginas web.</li> <li>❖ Tienen inseguridad pues existen muchos y se recomienda investigar al programador, opiniones, experiencias.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Licencias, propiedad intelectual.</li> <li>❖ Costos de licenciamientos y soporte periódicos.</li> <li>❖ Demasiada funcionalidad compromete uso y rendimiento.</li> <li>❖ No existe control sobre las nuevas versiones</li> <li>❖ Dependencia del fabricante</li> <li>❖ La integración no es trivial, los fabricantes no la garantizan.</li> <li>❖ Los componentes son cajas negras de unidades de programas, y el código de los componentes puede no estar disponible para los usuarios de dichos componentes.</li> </ul>
10	<b>Ejemplos</b>	<ul style="list-style-type: none"> <li>❖ .NET Framework.</li> <li>❖ Symphony.</li> <li>❖ Zend Framework.</li> <li>❖ Laravel.</li> <li>❖ Django.</li> <li>❖ Angular.</li> <li>❖ Entre muchos otros.</li> </ul>	<ul style="list-style-type: none"> <li>❖ WordPress SEO Yoast.</li> <li>❖ Tweet Old Post.</li> <li>❖ P3.</li> <li>❖ WooCommerce.</li> <li>❖ UpdraftPlus</li> <li>❖ WordPress</li> <li>❖ Backup Plugin.</li> </ul>	<ul style="list-style-type: none"> <li>❖ Java Beans.</li> <li>❖ Microsoft COM.</li> <li>❖ CORBA.</li> </ul>
11	<b>Fecha de creación</b>	Cada Framework tiene su propia fecha de creación, pero podemos considerar que los primeros frameworks fueron creados entre el año 1999 y 2000, siendo el serviet 2.2, el Tomcat 3.0.x y el ASP 3.0 los primeros en salir.	Los primeros plug-ins que se encontraron en el mercado fueron los VST. Fueron lanzados en 1999 por la compañía alemana Steinberg. Estos son usados hasta la fecha para integrar efectos de audio e	La idea de que el software debe estar componentizado (construido de componentes prefabricados) por primera vez llegó a ser prominente con el discurso de Douglas McIlroy en la conferencia de la OTAN sobre la ingeniería de software en

			instrumentos virtuales.	Garmisch, Alemania, 1968, titulado Mass Produced Software Components (componentes de software producidos en masa).
12	<b>Versiones que hay</b>	Existen una cantidad bastante grande de versiones debido a que cada pequeño cambio o errores arreglados pudieran considerarse como una nueva versión y debido a que también hay una vasta cantidad de frameworks diferentes.	Cada uno de los plug-ins que se encuentran en internet tiene diferentes versiones. Siempre están en constante actualización y buscando mejorar detalles.	Cada componente va evolucionando cuando se le agregan mejoras, puede suceder que el usuario desarrollador incorpore una versión más antigua a su sistema y tenga que reincorporar la versión nueva, lo que trae consigo que el sistema tenga que ser probado de nuevo.
13	<b>Lenguaje utilizado</b>	Los lenguajes en los frameworks son diferentes para cada quien. Cada uno puede manejar un único lenguaje o varios lenguajes debido a que estos son los que se utilizan para desarrollarlo.	La mayoría de los plug-ins utilizados en páginas web están escritos en JavaScript y PHP y son implementados con HTML, hojas de estilo CSS.	Los componentes proporcionan una interfaz API que define los métodos que pueden ser usados por el usuario.
14	<b>Empresa desarrolladora</b>	Un ejemplo de empresas desarrolladoras pueden ser Microsoft, Symphony Community, Zend Technologies, Taylor Otwell, Django Software Foundation, entre otros.	WordPress contiene una gran cantidad de plug-ins. Dentro de WordPress existen empresas dedicadas a los plug-ins, como: <ul style="list-style-type: none"> <li>❖ WordSuccor Ltd.</li> <li>❖ Varshyl.</li> <li>❖ Macrosoft Inc.</li> <li>❖ WPWeb Infotech Private Limited.</li> <li>❖ BaymediaSoft</li> </ul>	Empresas como OMG (CORBA), Sun Microsystems (JavaBeans) y Microsoft (COM), son algunos ejemplos.

15	<b>Rendimiento</b>	Algunos frameworks contienen análisis de rendimiento donde puedes observar los recursos que está utilizando y cómo pueden afectar a tu sistema. Un ejemplo es .NET Framework.	No presentan problemas de rendimiento si se optimiza bien la combinación de plug-ins. Incluso existen tipos de plug-ins para mejorar el rendimiento de los sistemas.	En el desarrollo de software basado en componentes la primera actividad es la de “Evaluación de componentes” donde el desarrollador hace las evaluaciones necesarias con el fin de encontrar los componentes que satisfacen los requerimientos del sistema y cumplan con los estándares de calidad.
16	<b>Curva de aprendizaje</b>	Aprender a utilizar un framework puede ser un proceso complejo que requiere de tiempo y práctica para poder dominar todas sus opciones. Por eso es importante conocer el nivel de dificultad de aprendizaje de uso de un framework antes de elegirlo.	Usar plug-ins es bastante sencillo y no es necesario mucho tiempo y práctica. Sin embargo, se deben tener conocimientos sobre cómo personalizarlos ya que todos son diferentes.	Los fabricantes no suelen documentar de forma adecuada sus componentes y no suelen tener ninguna especificación en sus interfaces. Por lo que es difícil para los clientes desarrolladores conocer los detalles la especificación del componente, información sobre la interfaz, protocolos de comunicación, características de implantación, etc.
17	<b>Seguridad</b>	Los desarrolladores de frameworks trabajan hasta en los mínimos detalles para que el producto que ofrecen permita crear sitios seguros y con estándares válidos.	Para asegurarnos de que usaremos plug-ins seguros es necesario conseguirlos de repositorios oficiales, cualquier otra fuente puede ser un foco rojo para no usarlo en nuestro sistema.	Para la selección de componentes a un sistema se proponen modelos de calidad como ISO/IEC 9126, para tener la seguridad que los componentes estarán libres de fallos en todo momento.

18	<b>Requisitos</b>	Para poder utilizar .NET Framework es necesario contar con un procesador de 1 GHz, RAM 512 MB y espacio mínimo de 4.5 GB.	Los plug-ins de IBM suelen ser bastante ligeros y no presentan problemas en computadoras antiguas. Sin embargo, en servidores web se recomienda contar con mínimo 350 MB de espacio libre.	Los fabricantes de componentes deben de especificar la información detallada sobre el componente y los requisitos necesarios para su implantación.
19	<b>Plataforma</b>	Principalmente se centra en escritorio por la facilidad de desarrollo.	Los plugins se encuentran en básicamente todas las plataformas, principalmente en web y escritorio.	Los componentes proporcionan un servicio sin importar el lugar donde el componente se esté ejecutando, y sin importar su lenguaje.
20	<b>Adaptabilidad</b>	Los frameworks no se adaptan a proyectos creados porque pueden presentar problemas con compatibilidad de código. Es mejor usarlos en proyectos nuevos.	Los plug-ins, al contrario de los frameworks, se pueden implementar fácilmente en proyectos ya avanzados o incluso terminados.	Los componentes en un sistema son genéricos y pueden ser reemplazados por otros que ofrecen funcionalidades similares o por nuevas versiones de los mismos.

Fuentes de consulta:

Diseño Web & Marketing Online. (2020, 8 abril). *Qué es un framework y para qué se utiliza*. dipe. <https://blog.dipe.es/blog/que-es-un-framework-y-para-que-se-utiliza/>

Agencia de Marketing Online NeoAttack. (2020, 27 agosto). *¿Qué es un Framework y para que sirve? - Neo Wiki*. NeoAttack. <https://neoattack.com/neowiki/framework/>

*¿Qué es un plugin y para qué se usa?* (2021, 17 septiembre). IONOS Digitalguide. <https://www.ionos.mx/digitalguide/servidores/know-how/que-es-un-plugin/>

Muente, G. (2020, 8 enero). *Guía completa del Framework: qué es, cuáles tipos existen y por qué es importante en Internet*. Rock Content. <https://rockcontent.com/es/blog/framework/>

*Plug in: Todo lo que necesitas saber*. (s. f.). Ryte Wiki. Recuperado 6 de octubre de 2021, de <https://es.ryte.com/wiki/Plug-In>

*¿Qué es un plugin y para qué se usa?* (2020, 9 octubre). IONOS Digitalguide. <https://www.ionos.mx/digitalguide/servidores/know-how/que-es-un-plugin/>

Ryte Digital Marketing Wiki. (s. f.). *¿Qué es un Framework?* Recuperado 6 de octubre de 2021, de <https://es.ryte.com/wiki/Framework>

Tagle, J. (2020, 1 mayo). *¿Qué es un plugin y para qué funciona?* WP Avanzado. <https://wpavanzado.com/que-es-un-plugin/>



*¿Afectan los plugins al rendimiento de WordPress? Datos de prueba reales.*  
(2020, 25 marzo). Wordpressbits. <https://wordpressbits.com/afectan-los-plugins-al-rendimiento-de-wordpress-datos-de-prueba-reales/>

Microsoft. (2017, 30 marzo). *Rendimiento de .NET Framework*. Microsoft Docs. <https://docs.microsoft.com/es-es/dotnet/framework/performance/>

Coppola, M. (2021, 28 enero). *Qué es un plugin, para qué sirve y cómo instalarlo*. hubspot. <https://blog.hubspot.es/marketing/que-es-plugin>

Tagle, J. (2015, 3 febrero). *Gastos necesarios para comenzar con Wordpress*. WP Avanzado. <https://wpavanzado.com/gastos-para-comenzar-wordpress/>

Bravo, L. (2018, 5 septiembre). *Framework o librerías: ventajas y desventajas*. tiThink Technology. <https://www.tithink.com/es/2018/08/29/framework-o-librerias-ventajas-y-desventajas/>

Colectiva, N. (2018, 4 septiembre). *Que es un Framework, Historia y Más Detalles*. Blog Nube Colectiva. <https://blog.nubecollectiva.com/que-es-un-framework-historia-y-mas-detalles/>

Flores J., (2009). Técnicas de adaptación de componentes. [Tesis de licenciatura, Centro de Investigación en Matemáticas]. Repositorio CIMAT. <https://cimat.repositorioinstitucional.mx/jspui/handle/1008/185>

Sommerville, I. (2011). *Ingeniería De Software* (9.a ed.). Pearson Educación. <https://tognuhome.files.wordpress.com/2020/10/ingenierc3ada-del-software-ian-sommerville.pdf>

Pintado P., Farfán M., Rojas R. (2008). Investigación del proceso de Ingeniería de Software basado en componentes (ISBC) y su aplicación a un caso práctico. [Trabajo de fin de grado, Universidad del Azuay]. Repositorio institucional Universidad del Azuay. <http://dspace.uazuay.edu.ec/handle/datos/2330>

Ariza, M. & Molina, J. (2004). Introducción y principios básicos del desarrollo de software basado en componentes.

[https://www.academia.edu/8254931/Componentes\\_de\\_Software\\_1\\_INTRODUCCI%C3%93N\\_Y\\_PRINCIPIOS\\_B%C3%81SICOS\\_DEL\\_DESARROLLO\\_DE\\_SOFTWARE\\_BASADO\\_EN\\_COMPONENTES](https://www.academia.edu/8254931/Componentes_de_Software_1_INTRODUCCI%C3%93N_Y_PRINCIPIOS_B%C3%81SICOS_DEL_DESARROLLO_DE_SOFTWARE_BASADO_EN_COMPONENTES)