UNIVERSIDAD NACIONAL DE MOQUEGUA

FACULTAD DE INGENIERIA Y ARQUITECTURA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA



INFORME DE EXAMEN PROFESIONAL

DESARROLLO DE APLICATIVOS WEBS, BENEFICIOS Y DIFERENCIAS AL UTILIZAR UN FRAMEWORK O UN CMS (CONTENT MANAGEMENT SYSTEM)

PRESENTADO POR:

BACH. HENRRY RENZO TICONA CONDORI

PARA OPTAR EL TITULO PROFESIONAL DE:

INGENIERO DE SISTEMAS E INFORMATICA

MOQUEGUA-PERÚ

2022

UNIVERSIDAD NACIONAL DE MOQUEGUA

FACULTAD DE INGENIERIA Y ARQUITECTURA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

INFORME DE EXAMEN PROFESIONAL

PRESENTADO POR:

BACH. HENRRY RENZO TICONA CONDORI

PARA OPTAR EL TITULO PROFESIONAL DE:

INGENIERO DE SISTEMAS E INFORMATICA

_____

BACH. HENRRY RENZO TICONA CONDORI

_____

MSC. HUGO EULER TITO CHURA

ASESOR

MOQUEGUA -PERÚ

UNAM
UNIVERSIDAD NACIONAL DE MOQUEGUA

UNIVERSIDAD NACIONAL DE MOQUEGUA
FEDATARIO

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

Folio Nº 006

ACTA DE TITULACIÓN

I. Modalidad:

Examen Profesional

II. Documento de aprobación:

Resolución Comisión Organizadora N° 636-2022-UNAM

Fecha: 19 de agosto 2022

III. Título de Informe de Examen Profesional:

"Desarrollo de Aplicativos Webs; beneficios y diferencias al utilizar un Framework o un CMS (Content Management System)".

Bachiller: Henrry Renzo Ticona Condori

Asesor Principal: M.Sc. Hugo Euler Tito Chura

Co-Asesor: .....

IV. Se dictamina con las siguientes notas:

Examen Escrito: 15.20 Entrevista: 3.07 Promedio: 18

Promedio en letras: Dieciocho por lo que: Sí [X] No [ ]

Opta el Título Profesional de INGENIERO DE SISTEMAS E INFORMÁTICA.

Siendo las 10:08 horas, del 04 de octubre del año 2022 se procede con la firma en conformidad de lo dictaminado.

M.Sc. Hugo Euler Tito Chura
Presidente del Comité de Grados y Títulos
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

M.Sc. Victor Yana Mamani
Director
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

M.Sc. Hugo Euler Tito Chura
Asesor
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

Bach. Henrry Ticona Condori

DEDICATORIA

Principalmente a mi Madre y familia, quienes han creído en mi persona siempre, dándome ejemplo de superación, respeto y sacrificio ante todo las adversidades que se me presente de acá hacia adelante en mi formación profesional, a todos ellos dedico el presente trabajo. Espero contar siempre con su valioso en incondicional apoyo.

AGRADECIMIENTO

Agradezco principalmente a Dios quien me ha guiado y me dio la fortaleza para seguir adelante, a mi asesor quien es la persona que me ha guiado y apoyado en la realización de este trabajo y a todas las personas quienes han contribuido a la realización de este trabajo.

RESUMEN

la complejidad durante el periodo de elaboración de aplicaciones web en la actualidad, se ven comprometidos desde el inicio a entregar un producto funcional en tiempos reducidos, manteniendo la calidad del software. Para permitir a los programadores satisfacer las necesidades a la hora de desarrollo, por tal motivo se ha promovido como solución el uso de una arquitectura conocidos como Frameworks para el desarrollo web, con el objetivo principal de reducir drásticamente tiempos de trabajo y minimizar errores.

Uno de los aspectos que ha influido en la evolución de internet en estos últimos años ha sido la aparición de múltiples herramientas de gestión de contenidos (CMS), que han permitido a una gran parte de la población publicar contenidos fácilmente y hacerlos visibles en la web.

En la actualidad existen diferente Frameworks y CMS (Content Management System), pues cada uno de estos dan diferentes soluciones a problemas, por lo que se convierte en un arduo proceso al momento de la selección. Es por esto que en este trabajo se dará a conocer claramente los beneficios y las diferencias entre un Framework y un CMS, para tomar una buena decisión al momento de ejecutar un proyecto.

Palabra clave: Framework, CMS, Software, internet.

ABSTRACT

The complexity during the period of elaboration of web applications nowadays, they are committed from the beginning to deliver a functional product in reduced times, maintaining the quality of the software. To allow programmers to meet the needs at the time of development, for this reason it has been promoted as a solution the use of an architecture known as Frameworks for web development, with the main objective of drastically reducing work times and minimizing errors.

One of the aspects that has influenced the evolution of the Internet in recent years has been the emergence of multiple content management tools (CMS), which have allowed a large part of the population to easily publish content and make it visible on the web.

Currently there are different Frameworks and CMS (Content Management System), as each of these give different solutions to problems, so it becomes an arduous process at the time of selection. This is why this paper will clearly show the benefits and differences between a Framework and a CMS, to make a good decision when executing a project.

Keyword: Framework, CMS, Software, Internet.

ÍNDICE GENERAL

DEDICATORIA I

AGRADECIMIENTO II

RESUMEN III

ABSTRACT IV

INTRODUCCIÓN 1

CAPÍTULO I 2

ASPECTOS GENERALES 2

1.1 DESARROLLO DE APLICATIVOS WEBS, BENEFICIOS Y DIFERENCIAS AL UTILIZAR UN FRAMEWORK O UN CMS (CONTENT MANAGEMENT SYSTEM). 2

CAPÍTULO II 3

DESARROLLO DEL TEMA DEL EXAMEN PERSONAL 3

2.1 DEFINICIÓN DE FRAMEWORK 3

2.1.1 Utilidades de un framework. 4

2.1.2 Características de un Framework 5

2.1.3 Arquitectura MVC (Modelo Vista Controlador) 5

2.1.4 Tipos de Framework 9

2.1.5 Clasificación de frameworks 9

2.1.6 Framework más Conocidos 11

2.1.7 Ventajas del Uso de Framework 14

2.1.8 Desventajas del Uso de Framework 15

2.2 DEFINICION DE UN CMS (CONTENT MANAGEMENT SYSTEM) 16

2.2.1 Características de un CMS 17

2.2.2 Tipos de CMS 20

2.2.3 Tipos de Aplicaciones utilizando CMS 21

2.2.4 CMS para comercio electrónico 26

2.2.5 Ventajas y desventajas de un CMS 27

2.3 DIFERENCIA ENTRE FRAMEWORK Y UN CMS 29

CAPÍTULO III 31

ANTECEDENTES Y OBJETIVOS 31

3.1 Antecedentes Internacionales 31
3.2 Antecedentes Nacionales 32
CAPÍTULO IV 34
CONCLUSIONES Y RECOMENDACIONES 34
4.1 CONCLUSIONES 34
4.2 RECOMENDACIONES 35
BIBLIOGRAFÍA 36

ÍNDICE DE FIGURAS

FIGURA 1. DISEÑO ESTRUCTURAL DE UN FRAMEWORK..... 4

FIGURA 2. ESQUEMA DE LA ARQUITECTURA MVC 7

FIGURA 3. EJEMPLO DE INTERACCIÓN DEL PATRÓN MVC 8

FIGURA 4. ESQUEMA DE FLUJOS DE PROCESOS EN UN CMS..... 17

ACRÓNIMOS Y ABREVIATURAS

CMS:	Sistema de gestión de contenidos
MVC:	Modelo Vista Controlador
GPL:	General Licencia pública
WYSIWYG:	Lo que ves es lo que obtienes
URL:	Localizador de Recursos Uniforme
XLM:	Lenguaje de marcado extensible
HTML:	Lenguaje de Marcas de Hipertexto
CSS:	Hojas de Estilo en Cascada
PHP:	Preprocesador de Hipertexto
WAF:	Framework para Aplicaciones Web

INTRODUCCIÓN

En el mundo actual del desarrollador se presentan herramientas que nos facilitan la creación de aplicaciones webs como los frameworks (marco de trabajo), que son un conjunto normalizado de conceptos, prácticas y criterios para procesar un tipo de problemática específico, que sirve como referencia para revolver nuevos problemas de índole similar.

La utilización de frameworks de software libre, se propone como una solución tecnológica a un problema de desarrollo web, los frameworks de programación de código libre proponen esta solución. Por tal motivo, existen muchos frameworks proporcionadas por diferentes empresas, desarrolladoras de software, comunidades y/o individuos particulares con el fin de suplir este tipo de necesidades.

Uno de los momentos de evolución de un CMS, fue en la expansión de blog personal que se dio hace años, debido a la aparición de multitud de herramientas de código abierto y gratuitos que permite a cualquier persona crear aplicativos webs, que hasta entonces había estado restringido a gente con conocimientos técnicos y que se daba normalmente en los ámbitos de programadores.

Teniendo en cuenta lo mencionado anteriormente, este trabajo se enfocará en dar a conocer todos los beneficios de utilizar framework o un CMS. Esto tiene una ventaja para el programador y usuario en el momento de elegir una de estas opciones para la ejecución de un proyecto para las organizaciones.

CAPÍTULO I

ASPECTOS GENERALES

1.1 Desarrollo de Aplicativos webs, Beneficios y Diferencias al Utilizar un Framework o un CMS (Content Management System).

En el desarrollo de un sitio web, elegir entre un Framework o CMS (Content Management System) puede ser indeciso. Esta decisión debe ser bien informada, para lo cual debe comprender claramente la diferencia entre ambos. En este Informe se dará a conocer estas diferencias y las ventajas, desventajas de cada uno para poder decidir cuándo usar uno u otro según el proyecto que vayas a desarrollar.

CAPÍTULO II

DESARROLLO DEL TEMA DEL EXAMEN PERSONAL

2.1 Definición de Framework

Es un entorno de trabajo que nos ayuda a realizar aplicaciones webs profesionales, sólidas y dinámicas mediante un conjunto de paquetes, herramientas y beneficios que aligeran el proceso de desarrollo de un proyecto.

También podemos definirlo como un conjunto de archivos y directorios que facilitan la creación de aplicaciones, ya que incorporan funcionalidades ya desarrolladas y probadas, implementadas en un determinado lenguaje de programación.

“Conjunto de códigos predefinidos o bibliotecas que proporcionan funcionalidad común a toda una clase de aplicaciones. Puede ser visto como una base o esqueleto para construir aplicaciones o sitios web” (Trninic, 2010).

“Estructuras para la creación de aplicaciones web que permiten al programador desarrollar aplicaciones con interfaces interactivas y también para que la aplicación se ejecute en una máquina que está controlada por el proveedor de la aplicación” (Schwarz, 2013).

“Un framework es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado” (Gutierrez, 2007).

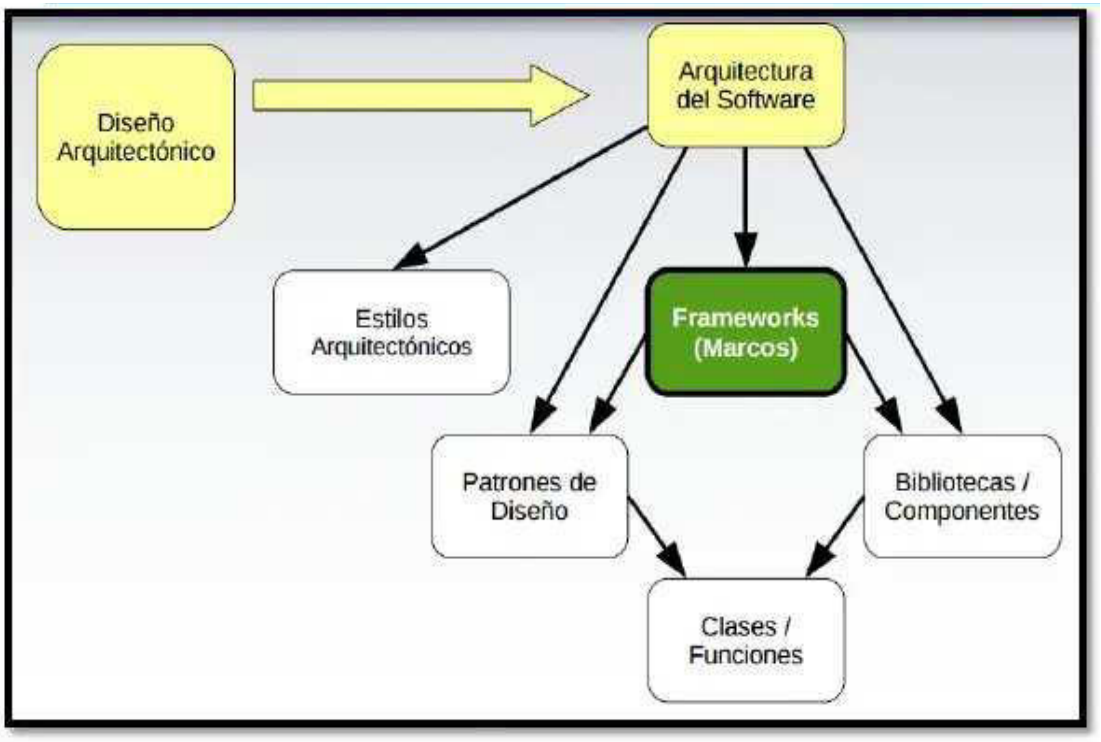
Un framework, podemos definirlo como un conjunto de componentes (por ejemplo, clases en java y descriptores y archivos de configuración en XML) que componen un diseño reutilizable que facilita y agiliza el desarrollo de sistemas Web.

2.1.1 Utilidades de un framework.

La idea de utilizar un framework a la hora de desarrollar aplicaciones es la de disminuir el tiempo de desarrollo, reutilizar código existente, y promover buenas prácticas de desarrollo como el uso de patrones o estándares. Bajo esa idea, el uso de frameworks define una filosofía de trabajo, donde el objetivo de un proyecto puede ser sintetizado en uno solo y es en su alcance, donde el desarrollador se centrará, dejando los detalles de operación a las estructuras del framework.

Figura 1.

Diseño estructural de un Framework



Nota: Gutiérrez, D. (2010). Frameworks y Componentes. Universidad de los Andes.

2.1.2 Características de un Framework

Se pueden mencionar las siguientes características:

- **Abstracción de URL y sesiones:** No es necesario manipular directamente las URL ni las sesiones, el framework ya se encarga de efectuarlo por sí mismo.
- **Controladores:** La gran mayoría de frameworks implementa una serie de controladores para gestionar eventos, esto lo realiza introduciendo datos mediante un formulario o el acceso a una página. Estos controladores suelen ser fácilmente compatibles a las necesidades de un proyecto específico.
- **Acceso a datos:** Estos incluyen las herramientas e interfaces necesarias para integrarse con herramientas de acceso a datos, en BBDD, XML, etc.
- **Autentificación y control de acceso:** Incluyen mecanismos para la identificación de usuarios mediante login y password, esto permite restringir el acceso a determinas páginas a determinados usuarios.
- **Internalización:** Son mecanismos para poder mostrar la aplicación en todos aquellos idiomas que consideremos oportunos (AcensTechnologies).
- **Extensibilidad:** Un marco puede ser amplio para proporcionar una funcionalidad específica. El framework, en general, no se supone que deba ser modificado, excepto en cuanto se trata de extensibilidad. Los usuarios puedan ampliar sus características, pero no deben ni necesitan modificar su código.

2.1.3 Arquitectura MVC (Modelo Vista Controlador)

En la mayoría de Frameworks webs actuales, es necesario entender este patrón de diseño conocido como Modelo-Vista-Controlador (MVC). Este patrón es una guía para el diseño de arquitecturas de aplicaciones que ofrecen una gran

interactividad con el usuario (Navarrete, 2017). La idea con este patrón es de trabajar, de manera más organizada y profesional, las aplicaciones en tres componentes diferenciados entre sí.

2.1.3.1 Modelo

En esta capa se almacenan archivos que contienen metodologías que hacen consultas lógicas a una tabla de la Base de Datos y también se puede realizar otras gestiones que sean necesarias en esta capa.

2.1.3.2 Controlador

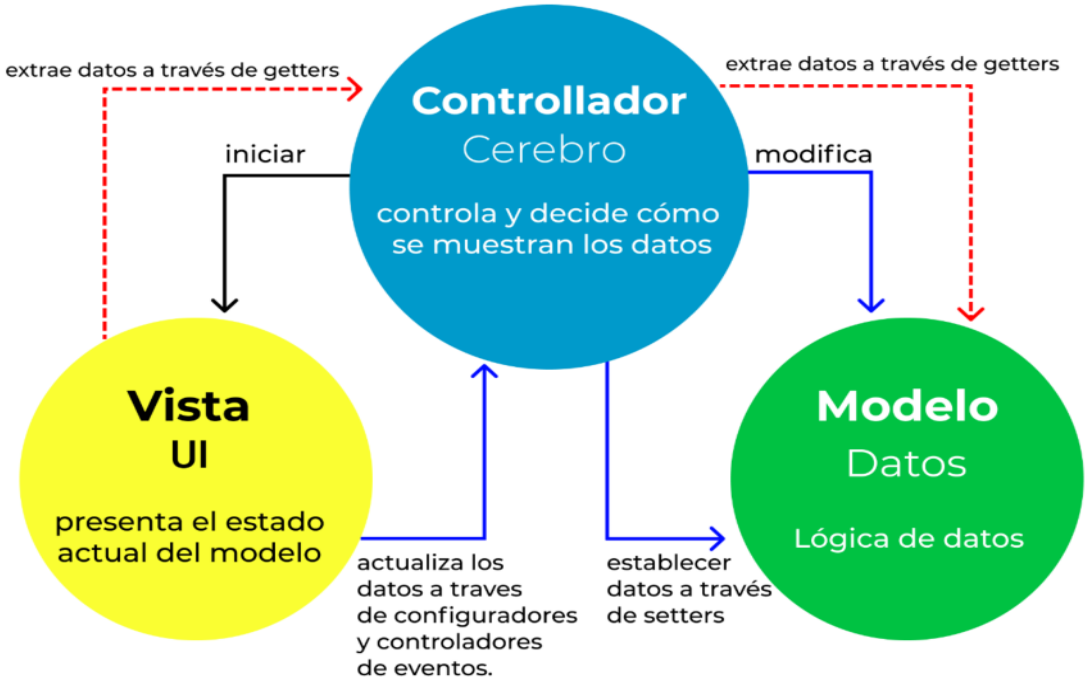
Esta capa es la encargada de recibir o encontrar los archivos que contienen código de una consulta de datos a la base de datos dinámico y estos son comunicados a la Vista.

2.1.3.3 Vista

En esta capa es donde se puede ubicar las páginas que serán vistas por los usuarios, todo lo que gestiona un controlador lo envía como vistas para ser visualizados en la interfaz.

Figura 2.

Esquema de la arquitectura MVC



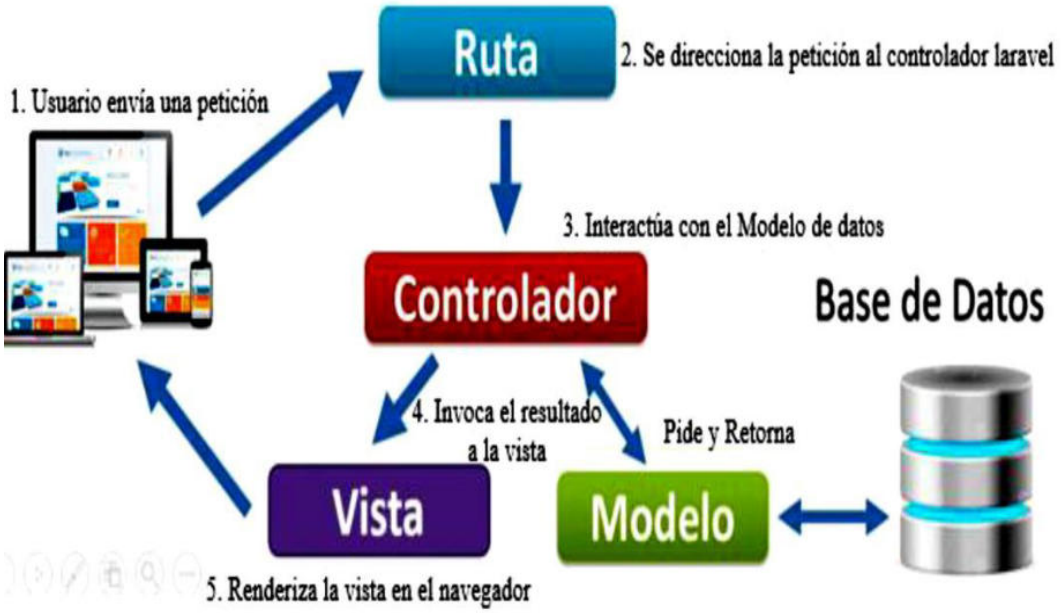
Nota: Rafael D. Hernández, El patrón modelo-vista-controlador

El significado de MVC fue conocido por primera vez por Trygve Reenskaug, quien lo planteó como una forma de desarrollar el GUI de aplicaciones.

"Trygve Mikkjel Heyerdahl Reenskaug (nació el 21 de junio de 1930) es un informático noruego y profesor digno de la Universidad de Oslo. Formuló el patrón modelo-vista-controlador (MVC) para el diseño de software de interfaz gráfica de usuario (GUI) en 1979 mientras visitaba el Centro de Investigación Xerox Palo Alto (Reenskaug, 2018)."

Figura 3.

Ejemplo de interacción del patrón MVC



Nota: Realizado por: Cabrera P., 2019

Adoptando la arquitectura de Modelo Vista Controlador, un framework al final busca no solo facilitar el trabajo del desarrollador, sino también, el desarrollo de aplicaciones más intuitivas para el usuario final, procurando que las interfaces, la lógica y los procesos sean tan prácticos como sea posible.

De esta manera, el desarrollador evita estar pensando en una serie de detalles que pueden distraerlo a la hora de desarrollar una aplicación que resulte cómoda para los usuarios y se centra en atacar necesidades reales sin tener que partir de supuestos de utilidad.

2.1.4 Tipos de Framework

Entre los tipos de Framework podemos mencionar:

2.1.4.1 Frameworks para Desarrolladores Front-end

Son los framework que facilitan el trabajo para las vistas o páginas que serán vistas por el usuario final o el público, este tipo de Framework se componen en su core con tecnologías como JavaScript, HTML y CSS. Podemos mencionar para esta categoría a Frameworks como Angular JS, React JS, VueJS, Bootstrap, Material Design entre otros.

2.1.4.2 Frameworks para Desarrolladores Back-end

Este tipo de Frameworks facilitan el trabajo con Lenguajes de Programación de lado del Servidor como Python, Ruby, PHP, Javascript del lado del servidor (Node JS), Java, etc. También ayudan a gestionar las Bases de Datos, el envío y procesamiento de datos, Cookies, Sesiones, etc.

En esta categoría podemos encontrar a Frameworks como Ruby, Django, Node JS, Spring, Laravel, React, Native JS, etc.

2.1.5 Clasificación de frameworks

Aunque los beneficios y los principios de diseño subyacentes de los frameworks son independientes del dominio en donde son aplicados, es útil clasificarlos según su alcance, así es posible definir:

2.1.5.1 Frameworks para la infraestructura de sistemas

Simplifican el desarrollo de infraestructuras de sistemas portables y eficientes, como sistemas operativos y frameworks de comunicación, y frameworks para

interfaces de usuario y herramientas para el procesamiento de lenguajes. Los frameworks para la infraestructura de Investigación previa sistemas son usados principalmente internamente dentro de una organización de software y no se encuentran disponibles para los clientes directamente.

2.1.5.2 Frameworks para la integración del Middleware

Comúnmente utilizados para integrar aplicaciones y componentes distribuidas. Están diseñados para facilitar a los desarrolladores a modular, reutilizar y escalar sistemas para trabajar mejor en un entorno distribuido.

2.1.5.3 Frameworks para empresas

Abarcan amplios dominios de aplicación y son el núcleo de las actividades de negocio de la empresa. Son costosos de desarrollar y/o comprar, pero esenciales para crear rápidamente software de alta calidad.

2.1.5.4 Frameworks para aplicaciones Web

“Un Framework para Aplicaciones Web (WAF) consiste de una plataforma modular reutilizable y semicompleta que puede ser especializada para producir aplicaciones web. Estas aplicaciones comúnmente son utilizadas por navegadores Web vía el protocolo HTTP. Un WAF incluye un conjunto de servicios y componentes esenciales para la construcción de sistemas sofisticados y ricos en funcionalidades” (Shan, 2006).

En la actualidad existen varios frameworks para el desarrollo de aplicaciones web, siendo en su mayoría open source. Esta característica permite a los desarrolladores disponer del código fuente, y con esto,

aprender de otros desarrolladores y reutilizar algún patrón en su trabajo.

2.1.6 Framework más Conocidos

Zend Framework (ZF): “Un framework de código abierto para desarrollar aplicaciones web y servicios web con PHP. ZF es una implementación que usa código 100% orientado a objetos. La estructura de los componentes de ZF es algo único, cada componente está construido con una baja dependencia de otros componentes” (Buenas prácticas en el desarrollo de aplicaciones con ZEND , 2016).

Estos utilizan PHPUnit para realizar pruebas de código y Travis CI como servicio de integración continua, ofrece una aplicación robusta de alto rendimiento MVC, también cuenta con un componente de formularios que son implementados en HTML, otros componentes a mencionar son de autenticación y permiso de usuarios. Dada la calidad de este framework, empresas como Google, Microsoft, y StrikeIron se han asociado con Zend para proporcionar interfaces de servicios web y otras tecnologías que desean poner a disposición de los desarrolladores.

Laravel: “Es un marco de aplicación web con una sintaxis expresiva y elegante. Un marco web proporciona una estructura y un punto de partida para crear su aplicación, lo que le permite concentrarse en crear algo increíble mientras nos preocupamos por los detalles” (Instalación – Laravel, 2011).

Este framework pretende combinar lo principal de otros frameworks como Ruby, ASP.NET, MVC y Sinatra. Está de más decir que utiliza el patrón MVC.

Utiliza ORM para sus bases de datos y cuenta con un sistema de migración, trabaja muy bien con SQLite, PostgreSQL, SQL Server y MySQL. Ejecuta sus pruebas de código con PHPUnit y también cuenta con un sistema amplio de plantillas.

Cake PHP: Estas aplicaciones web se realizan de forma rápida y simple requiriendo menos código, puedes construir rápidamente tus aplicaciones con funciones para generar código y así poder utilizarlos como prototipos, su configuración para conexión a base de datos es demasiado simple de esa forma te despreocupas de este aspecto y te dedicas netamente a hornear tus aplicaciones.

Utiliza un patrón MVC bastante limpio, de tal forma que puedes tener tu aplicación totalmente modular sin olvidar que también cuenta con sus respectivos componentes como el de autenticación de usuarios, validación de datos, almacenamiento en cache y muchos otros.

Symfony: Es un framework PHP construido de varios componentes independientes, que son creados por el proyecto Symfony, instruirse y aprender a programar con Symfony te ofrece una oportunidad para acceder a varios de sus proyectos que consta del framework Symfony para proyectos complejos para sitios web más sencillos, su código, y el de todos los componentes y librerías que incluye, se publican bajo la licencia MIT de software libre. Los componentes de Symfony son tan útiles y están tan probados, que proyectos tan gigantescos como Drupal, PHPBB y EZ Publish fueron construidos con dichos componentes.

CodeIgniter: Es un framework open source el cual tiene como objetivo permitir que los desarrolladores puedan realizar proyectos mucho más rápido que creando toda la estructura desde cero, proporcionando un amplio conjunto de bibliotecas para tareas comúnmente necesarias, de tal forma que permite

enfocarse en un proyecto minimizando la cantidad de código para una tarea determinada.
Su sistema está basado en el patrón MVC, contiene formularios y validaciones de datos, también cuenta con una manipulación de imágenes donde se puede recordar, redimensionar y rotar, soportando librerías GD, ImageMagick, y NetPBM. Cuenta con componentes de localización, paginación, cache de almacenamiento de página completa, URL amigables para motores de búsqueda y bibliotecas de funciones.

Yii Framework: Es un framework PHP de alto beneficio para desarrollo de aplicaciones web basado en la arquitectura MVC aplicando el principio de separación de preocupaciones (SoC), integrado con JQuery, Yii trae un conjunto de widgets AJAX como ser campos de autocompletado, vistas de árbol, cuadriculas de datos, etc. esto permite obtener una interfaz de usuario altamente eficiente y versátil. Además, Yii ha incorporado soporte de autenticación manejando el control de acceso basado en roles jerárquicos (RBAC).
Yii cuenta con muchas medidas de seguridad que ayudan a evitar ciertos ataques tales como SQL injection, cross-site scripting (XSS), request forgery (CSRF), cross-site y modificación de las cookies.

Kohana: Es un framework MVC (Modelo-vista-controlador) que proporciona un conjunto de componentes para la construcción de aplicaciones web, muchos de estos componentes incluyen herramientas de traducción, acceso a bases de datos, perfiles de código, validación, cifrado y muchos más. Es un framework orientado a objetos, por lo que está construido usando clases y objetos.
La extensión de nuevos componentes y complementos de nuevas bibliotecas es realmente fácil y utiliza la licencia BSD, por lo que se puede utilizar y modificar para fines comerciales.

Django: Es un framework realizado en Python de alto nivel que fomenta un desarrollo rápido, diseño limpio y pragmático. Creado por diferentes desarrolladores experimentados, estos se ocupan de gran parte de las molestias del desarrollo web, lo que permite que el desarrollador pueda concentrarse en escribir su aplicación sin necesidad de reinventar mucho código. Este Framework es gratis y de código abierto.

- Fue implementado para ayudar a los desarrolladores a llevar las aplicaciones desde el concepto hasta su finalización lo más rápido posible.
- La seguridad es muy importante y esto ayuda a los desarrolladores a evitar muchos errores de seguridad comunes.
- Algunos de los sitios web más concurridos aprovechan la capacidad de este Framework para escalar de forma rápida y flexible.

2.1.7 Ventajas del Uso de Framework

Entre las ventajas del uso de un Framework podemos mencionar los siguientes:

Organización de código y de archivos. Cuando instalas un framework en PHP, por defecto ya vienen ciertas estructuras en sus carpetas. Esperando que el programador solo siga algunos estándares y mantenga todo organizado de cierta manera.

Utilidades y Librerías. PHP es un gran lenguaje para desarrollo de aplicativos webs y te provee incontable número de herramientas y librerías.

Patrón MVC. Este tipo de arquitectura te permite un código más limpio y mantenible.

Menos código y más desarrollo. Escribirás menos código, lo cual significa menos tiempo escribiendo. No tendrás que declarar librerías de terceros cada vez para cada nuevo proyecto porque la mayoría de ellas vienen por default al instalar el framework.

Soporte de la comunidad. Todos los grandes Frameworks de PHP tienen grandes y activas comunidades detrás de ellos. Puedes hablar con otros desarrolladores, obtener ayuda, feedbacks y además dar tu ayuda a la comunidad.

Herramientas de Performance. Por defecto estas herramientas te ayudan con el caché, paginación, etc.

Facilita la colaboración. Cualquiera que haya tenido que luchar con el código fuente de otro programador o incluso el de uno mismo sabrá lo difícil que es entenderlo y modificarlo; por tanto, todo lo que sea definir y estandarizar va a ahorrar tiempo y trabajo a los desarrollos.

2.1.8 Desventajas del Uso de Framework

El uso de los framework también tiene sus limitaciones, por tal motivo, es bueno conocer e informarse antes de embarcarse en un proyecto. Las desventajas principales son:

Tiempo de aprendizaje. Te llevará algún tiempo conocer cómo funcionan los frameworks, pero la idea es que una vez que te habitúes a ellos, los siguientes proyectos tendrán una base más definida y su implementación será más rápida.

Exceso de líneas de código. Diversos autores y expertos en el desarrollo de aplicaciones apuntan que los framework utilizan muchas líneas de código para realizar ciertas acciones, ya que suelen incluir (código basura) que nos podríamos haber ahorrado.

Limitaciones. Cuando usas un framework hay partes de código que no puedes modificar, por ello hay que elegir uno que se adapte a lo que buscas. Asimismo, migrar a otro framework es complicado sin tener que reescribir todo el código.

Código público. Al encontrarse disponible el framework para todo el mundo, un hacker puede estudiar el código y encontrar vulnerabilidades, aunque es algo difícil, porque tendría que saber qué framework estás usando. Además, algunos Frameworks como Symfony son testados continuamente para reducir al máximo los bugs.

2.2 Definición de un CMS (Content Management System)

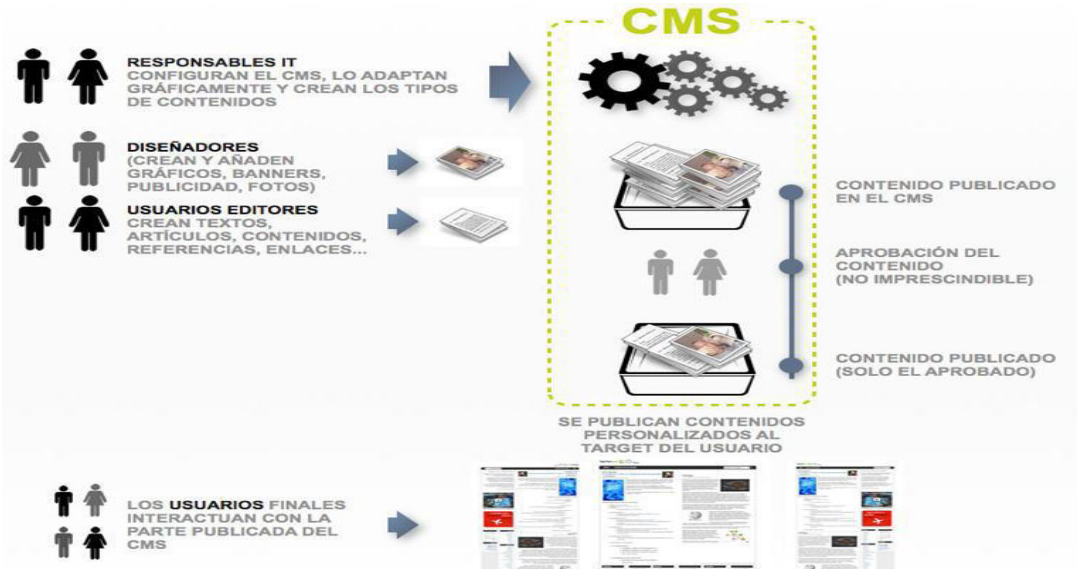
CMS son siglas de Content Management System, que se traduce directamente al español como sistemas de Gestor de Contenidos. Como su propio nombre indica, es un sistema que nos permite gestionar contenidos, permitiendo a los editores cambiar cualquier texto, video, imagen con una manera cómoda y sin tener que tocar el código fuente.

También se puede decir que un CMS es una herramienta que permite a un editor crear, clasificar y publicar cualquier tipo de información en una página web, de modo que el administrador simplemente actualiza la base de datos, incluyendo nueva información o editando la existencia.

"Un CMS es un conjunto de herramientas que por medio de software permiten introducir una estructura de soporte para facilitar la gestión de contenidos de sitios web, ya sea en internet o en una intranet, las hay en versiones gratuitas y propietarias, de acceso libre y restringido" (Cuerda, 2007).

Figura 4.

Esquema de flujos de procesos en un CMS.



Nota: MOSAIC - Tecnologías y Comunicación Multimedia.

2.2.1 Características de un CMS

Existen diferentes características que cualquier buen gestor de contenidos o CMS debe cumplir, entre características podemos mencionar las siguientes:

Facilidad en el uso del panel de administración: Uno de los aspectos cruciales en cualquier CMS es la facilidad del uso del panel de administrador.

Debemos tener en cuenta que este panel de administrador tiene al menos, dos tipos de usuarios, cada uno tendrá permisos diferentes para realizar algunas tareas.

Desde el punto de vista de la usabilidad, para estos usuarios un panel de administración debe ser:

- **Ser Intuitivo.** Las opciones deben ser autodescriptivas, esto significa que no es necesario de un manual para poder empezar a usarlo.
- **Proporcionar un editor de WYSIWYG (what you see is what you get).** De manera que no se necesiten conocimientos básicos de HTML, en estos editores es

conveniente poder activar el uso de etiquetas como cabeceras, listas, strong, etc.)

- **Permitir la previsualización.** Nos permite poder previsualizar el contenido antes de publicarlo, esto se realiza en la plantilla (tema) que tengamos activado, de modo que podemos ver el contenido tal como lo verá el usuario final.

Seguridad: En cualquier sistema informático siempre existe la posibilidad de que exista vulnerabilidades que son aprovechados por agentes y programas maliciosos. Cuando sucede estos casos, la comunidad desarrolladora de CMS solucionan el problema ofreciendo actualizaciones del software, por tal motivo es necesario actualizar los archivos de nuestros CMS periódicamente para evitar posibles daños que afecten al sitio web.

Separación de contenido y presentación: Esta característica es una de las más importantes debido a que el CMS debe encargarse de la gestión de contenidos y la estructura, pero el nivel de presentación debe estar totalmente separado, normalmente en forma de temas.
Un tema está construido por unas plantillas en algún lenguaje de programación como por ejemplo PHP y varios archivos CSS vinculados.

Extensibilidad (módulos): Un CMS tiene la capacidad de aumentar las funcionalidades del sistema con el menor costo de desarrollo posible. En la mayoría de estos casos se consigue mediante la instalación de extensiones llamados módulos o plugins.
En los CMS más usados como WordPress o Drupal, el número de extensiones se cuenta por miles y esto nos

permite ampliar enormemente el número de funcionalidades de los mismos.

Escalabilidad: Se define como la característica de mantener la capacidad del servicio a medida que la carga sobre el sistema aumenta. Esta carga se mide en número de visitas simultáneas como son el número de peticiones, número de páginas visitadas y el número de visitas únicas.

Accesibilidad: Otra característica que tiene los CMS es que tiene un grado alto de accesibilidad, por lo tanto, nuestro sitio web puede ser usado por cualquier persona en cualquier momento.

Es importante que todos los contenidos estén bien etiquetados para que puedan ser entendidos por lectores de pantalla u otros dispositivos.

Estabilidad: Esto se refiere al que sistema es tolerante a fallos o errores, es decir, que no generará ni producirá errores si se hace un uso correcto del mismo. Para asegurarnos que nuestro CMS sea estable debemos considerar los siguientes puntos.

- El CMS elegido debe contar con una base de usuarios y desarrolladores suficientemente grandes
- Debemos usar un CMS con versión de etiqueta estable y no una versión beta o alfa que todavía están en pruebas.
- El servidor que estamos usando cumple con los requisitos mínimos de instalación.

Facilidad de instalación y mantenimiento: Es un factor esencial de un CMS que permite la facilidad al momento de la instalación en un servidor y además esta facilidad se debe reflejar al momento de realizar el mantenimiento respectivo.

Las actualizaciones de programa o extensiones deben ser fáciles de realizar, esto permite que el código a lo largo del tiempo se vaya actualizando.

Interconectividad: Es la capacidad de un CMS de conectarse a otros sistemas de una organización (base de datos existentes), nos permite:

- Tener acceso instantáneo a base de datos compartidos.
- Optimización de recursos humanos.
- Reducción de presupuesto (dinero, tiempo).
- Administración de la red.
- Poder compartir recursos.

2.2.2 Tipos de CMS

Cada CMS tiene unos requerimientos técnicos diferentes (plataformas tecnológicas), están especializados según su función específica y tienen licencia de uso determinado. Según esta premisa podemos clasificarlos:

- **Según su plataforma:** En un servidor web tenemos normalmente tres componentes, servidor de páginas web, lenguaje de programación y servidor de bases de datos.
  La mayoría de los servidores de base de datos relacionales funcionan en SQL, un lenguaje de base de datos bastante estándar, de manera que a menudo es posible cambiar de servidor de datos sin grandes cambios en el código de CMS.
  Algunos lenguajes de servidor existentes sobre los cuales existen variados CMS ASP.NET, JAVA, PHP, RUBY ON RAILS, PYTHON, etc.

- **Según la propiedad de código:** Según esta aproximación podemos dividir las aplicaciones de CMS en dos grupos:
  - **Código abierto.** Son las que permiten el acceso al código fuente de manera que, si es necesario, se podrá modificar cualquier fragmento o incluso crear nuevas aplicaciones (nuevos CMS) a partir de la versión original.
  - **Código privado.** Solo su propietario o empresa creadora puede desarrollar o modificar el código base de la aplicación.

- **Según el tipo de aplicación.** En el momento de elegir un CMS, la primera elección se define por el tipo de aplicación que queremos desarrollar. Esta debe ajustarse a la funcionalidad del proyecto, es crucial para el éxito de su implementación.
  Existen CMS para distintas necesidades como:
  - Creación y administración de blogs.
  - Portales de empresas.
  - Entornos educativos.
  - Redes sociales.

2.2.3 Tipos de Aplicaciones utilizando CMS

A continuación, vamos a listar unos CMS especializados en implementación de aplicaciones de diferentes tipos:

OpenCms: Es un gestor de contenidos de código abierto desarrollado por la compañía Alkacon Software que está implementado sobre la plataforma en java (J2EE) / XML.

Como la mayoría de los CMS, tiene las funcionalidades básicas que son la administración vía web, gestión de activos (PDF, enlaces externos, etc.), editor de páginas WYSYWYG, control de versiones de contenidos, roll-back, vista previa de páginas,

gestión de usuarios, API de java, publicación de contenidos (estático y dinámico), implementación de módulos y función de búsqueda.

Ventajas

- Pensado para funcionar sobre muchos servidores webs.
- Posibilidad de tener páginas estáticas y dinámicas.
- Extensión mediante módulos opcionales.
- Facilidad de instalación.

Desventajas

- Muy poca documentación y de una buena comunidad de desarrolladores.
- OpenCms solo proporciona una funcionalidad limitada de flujo de trabajo.
- Como la mayoría de CMS, las funcionalidades se pueden ampliar mediante la instalación de módulos o extensiones.

Joomla: Es un CMS de código abierto con licencia GPL (general public license), basado en PHP / MySQL, también es conocido por su flexibilidad y adaptabilidad para las diferentes necesidades.

Este CMS cuenta con una gran comunidad de usuarios y desarrolladores que van actualizando las extensiones como son los componentes, módulos, plantillas, plugins y lenguajes.

Entre las funcionabilidades básicas tenemos:

- Cacheo de páginas.
- Pensado para optimizar los indexados en buscadores.
- Versiones para imprimir de todas las páginas.
- Encuestas.
- Calendarios.
- Generación de blogs
- Fórums.

Drupal: Es un CMS de código abierto con licencia GLP basado den PHP / MySQL, esta comunidad dedicada a este CMS es muy amplio, lo que permite que existe mucha documentación dando soporte a los problemas con su actualización a sus módulos.

Entre sus funcionalidades básicas son las que dan los módulos del núcleo (core), los que se instalan automáticamente sin necesidad de añadir módulos:

- Acceso a estadísticas.
- Búsqueda avanzada en el sitio.
- URL limpias.
- Blog (libro, fórums y encuestas).
- Sistema de menús multinivel.
- Soporte para múltiples usuarios.
- Soporte para múltiples áreas con la misma instalación.
- Notificaciones de actualizaciones de seguridad.
- Perfiles de usuarios.
- Roles de usuarios.
- Herramientas de workflow.
- Multiidioma.

Ventajas

- Plataforma altamente testeada.
- Una misma instalación para varios sitios.
- Potentes sistemas de plantillas, CSS plantillas puede ser fácilmente convertido a Drupal.
- Se adapta bien a todas las aplicaciones no solo en gestión de contenidos, sino también comunidades.

Desventajas

- El panel de administración no es demasiado intuitivo.
- Es necesario buscar y probar en la lista de miles de módulos disponible.
- Se debe tener cuidado con el gran número de módulos disponibles, ya que muchos de ellos no son estables.

- Si no cumple perfectamente los requerimientos de nuestro proyecto, puede ser difíciles para la adaptación.

WordPress: Es una de las plataformas CMS más utilizadas en blogs y también en proyectos web más generales. Desde el principio se planteó como un sistema para que los autores de blogs pudieran administrar el contenido de una manera sencilla.

Entre las funcionalidades básicas podemos mencionar:

- Activación sencilla de temas.
- Plugins que añaden funcionalidades.
- Instalación y actualización de temas y plugins directamente desde el panel de administración (sin tener que descargar archivos).
- Múltiples categorías para los artículos.
- Actualización remota de blog.
- Definición de diferentes tipos de contenidos.

Ventajas.

- Enfocado a la sencillez de su uso, el panel de administración es muy intuitivo.
- Optimizado para buscadores (SEO).
- Las páginas de WordPress pueden tener “padres” de manera que se puede crear una jerarquía.
- Existen infinidades de plugins que añaden funcionalidades.

Desventajas.

- No es la mejor opción para un sitio de miles de páginas paginas debido a su rendimiento.
- Aunque es muy flexible, sin las necesidades del sitio van más allá de las propias de WordPress, puede ser problemático.

Tiki Wiki: Es otro CMS enfocado a wikis de código abierto, con licencia GLP y basado en PHP. Para diferenciarse del resto pretende cubrir otras áreas de uso como webs genéricos.

Entre las funcionalidades básicas tenemos:

- Posibilidad de crear dibujos mediante un applet de java.
- Sondeos, encuestas y cuestionarios.
- Chat, mapas y gráficos.
- Capacidad crear diferentes tipos de contenidos como artículos, foros, boletines, blogs, galerías de imágenes y por supuesto wikis.
- Rastreadores.
- Integración de sistemas de correos.
- Calendario.
- Versión para móviles, Tiki móvil.
- Sistema de categorías.

Ventajas.

- No solamente es un CMS para wiki, sino también aporta varias herramientas útiles para el trabajo colaborativo, se podría decir que este CMS es más completo en todos los sentidos.
- Puede funcionar con un gran número de servidores de base de datos.

Desventajas.

- El rendimiento y escalabilidad necesitan mejorar.
- Este CMS ofrece tantas funcionalidades que es difícil que todas ellas se pueden ajustar bien a nuestros requerimientos.
- La gran cantidad de funcionalidades que implementa hace difícil la traducción de todas ellas a los diferentes idiomas.

Moodle: Uno de los CMS para e-learning más usados y con una gran comunidad de soporte es Moodle, también basado en PHP / MySQL y con licencia GPL. La comunidad de desarrolladores de Moodle ha permitido que este CMS este traducido a infinidades de idiomas.

- Entre las funcionalidades básicas son:
- Tareas y calificaciones para los alumnos.
- Consultas (funcionan como una votación).
- Foros exclusivos para profesores, alumnos y público en general.
- Cuestionarios, definir preguntas.
- Recursos, contendido digitales en formato Word, PowerPoint, video, etc.
- Encuestas.

Ventajas.

- Existe una gran comunidad de Moodle, lo que permite que exista una gran cantidad de extensiones y que muchos problemas se puedan solucionar buscando información en internet.
- Es muy configurable gracias a la presencia de muchos plugins.
- Desventajas.
- No existe un sistema de avisos general para todos los cursos, es necesario ir uno por uno para verlos.
- El control de los grupos es en el ámbito de curso.

2.2.4 CMS para comercio electrónico

Otra clasificación muy importante de los CMS es las que permiten implementar tiendas virtuales o comercio electrónico.

Magento: Es el más conocido de todos los sistemas de gestión de contenido que están orientados al comercio electrónico. Este CMS está considerado como la primera opción al

momento de pensar en realizar un proyecto de comercio electrónico, debido a su funcionalidad, robustez y arquitectura.

Prestashop: Es un CMS específico para la creación de tiendas virtuales, pero por el momento es más utilizado en España y Francia, debido a que nivel mundial decae su popularidad debido a que los plugins son de pagos y su arquitectura es demasiado compleja para su manipulación.

Shopify: En este caso nos encontramos con un CMS de comercio electrónico, que se ofrece como servicio. Es muy popular en todo el mundo como los anteriores CMS nombrados, es instalado en nuestros servidores, así podemos evitar ponernos límites y ahorrar problemas de migración en el futuro.

2.2.5 Ventajas y desventajas de un CMS

2.2.5.1 Ventajas

- Un CMS permite a la Organización concentrarse en la creación de contenido o en la función específica deseada para el sitio web, en lugar del diseño o el desarrollo de la herramienta, que habitualmente llevaría gran parte del tiempo.
- La creación de perfiles de usuarios en la gran mayoría de CMS son creados sin demasiada dificultad, esto permite que a estos perfiles se les pueda dar permisos específicos según sea el caso, como por ejemplo editar y publicar contenido en la web.
- Toda la información queda alojada en un solo lugar centralizado
  (base de datos), controlado por la propia organización. Esto permite tener un potencial en el

momento de interconectar esta información con otros sistemas de la organización (intranet).

- La búsqueda de información se puede realizar filtrándose por fecha, tipo de contenido, autor o por cualquier otro campo insertado.
- La programación de contendido se puede realizar en una fecha y hora determinada.
- Se pueden ampliar las funcionalidades como por ejemplo calendarios, foros, galerías de fotos, etc. Estas funcionalidades vienen incluidas en los CMS, estos se activan directamente en los plugins, extensiones o módulos según la necesidad del caso.
- Dado que el contenido y la presentación están separados, cambiar la apariencia (look-and-feel) del sitio web es mucho más fácil que usando un sistema hecho a medida. Esta característica es una de las más importantes que cumple cualquier CMS (HTML y CSS).

2.2.5.2 Desventajas

- Aunque se pueda afrontar cualquier proyecto con apenas desarrollo, cualquier CMS a menudo requiere mucho tiempo de configuración, adaptación e integración de todo el contenido.
- Existe una curva de aprendizaje inicial en el equipo que se encarga de actualizar los contenidos.
- Un CMS no convierte malos contenidos webs en buenos, ni convierte un mal escritor o editor en uno bueno. Los contenidos deben estar correctamente escritos. El CMS solo se encarga de gestionar los contenidos, pero no puede corregirlos.
- Migrar un CMS de una plataforma a otra puede ser complejo y costoso en tiempo y dinero. Por tal

motivo se tiene que anticipar a nuevas necesidades para poder hacer las transiciones de una manera gradual.

- Comúnmente, una persona o más de la organización debe convertirse en el webmaster del CMS para adoptar la responsabilidad de su mantenimiento.

2.3 Diferencia entre framework y un CMS

Luego de dar a conocer anteriormente las definiciones, tipos, ventajas y desventajas tanto de un Framework y un CMS, podemos observar diferencias entre estos que se debe de tomar en cuenta al momento de iniciar un proyecto.

Framework

1. Enfocados a la programación, el desarrollador utiliza las librerías de código que permite crear las aplicaciones.
2. Es un software que contiene funcionalidades genéricas que pueden ser modificados por código adicional escrito por el desarrollador.
3. Aprender a manejar un framework es mucho más difícil debido a que se debe tener conocimiento en programación.
4. Un framework permite organizar el código para que el proceso de desarrollo de aplicaciones sea más simple y flexible, pudiendo diseñarse para adaptarse a los requisitos del Proyecto.
5. Los tiempos de desarrollo son más largos, esto se debe a las funcionalidades y características que desea para su sitio web, lo que también genera un costo más elevado.
6. Códigos propios y personalizados, la gran mayoría de los framework vienen con funciones integradas como inyección Cross-Site Scripting que permite el cifrado y protección contra ataques.

CMS

1. La opción programación o codificación es opcional, debido a que la configuración se realiza a través del panel de administración.
2. Las funcionalidades se añaden con extensiones o módulos que se activan visualmente.
3. Trabajar con un CMS es más fácil, debido a que se tiene un panel de administrador que son fáciles de entender y navegar.
4. Existe una gran cantidad de comunidad dedicada a las actualizaciones, mejoras, esto permite dar soporte a los CMS.
5. Los gastos y los tiempos para el desarrollo de estos CMS son más económicos y cortos.
6. Son de código abierto, esto significa que está disponible para todos, por ende, la vulnerabilidad de la seguridad puede ser accedidos por piratas informáticos.

CAPÍTULO III

ANTECEDENTES Y OBJETIVOS

3.1 Antecedentes Internacionales

Martínez, Camacho & Biancha (2010), en su Artículo propone un Diseño de Framework web para el Desarrollo Dinámico de Aplicaciones. Presentando avances del proyecto que plantea el diseño y creación de un Framework web firme y eficiente, implementado con herramientas de software libre, esto ayudara en el desarrollo de aplicaciones web corporativas a través de asistentes, con su estructura general (seguridad, sesiones y navegación) y las diferentes páginas de gestión, a partir del modelo de datos de la aplicación.

Bojorque (2008)**,** en su artículo propone que tener una "Página Web", "Sitio Web", "Portal Web", para las empresas y organizaciones, ya no es un asunto únicamente informativo, ahora se trata de un tema Institucional de imagen y prestigio.
Considerando que es bastante sencillo, tener un sitio Web, con dominio y correo electrónico propio, hoy en día es un requisito indispensable, para toda empresa. Esencialmente es aquí donde los CMS han servido como herramientas de trabajo para contar con Portales Web completos en cuestión de días, lo que significo ahorro en tiempo y dinero, con características y cualidades que el propio usuario no imaginaba que podía tener.

Goodwin, Burford, Bedard & Carrigan (2006), el objetivo de este documento es describir la implementación sistemática de un CMS en las Bibliotecas de Texas A&M. El equipo de implementación de la web (WIT) se formó para incluir un grupo diverso de personas de todas las áreas de la biblioteca y se le encargó la responsabilidad de la gestión general del sitio web de las bibliotecas universitarias.

El uso de un CMS para crear la presencia web de la biblioteca es una empresa importante y costosa que requiere una supervisión coordinada de la gestión. También presenta oportunidades para reconsiderar la estructura organizativa y la cultura de la biblioteca.
Este artículo describe una estrategia de gestión que implica a todas las áreas de la organización, fomenta el trabajo en equipo, promueve la innovación y se mantiene centrada en las prioridades de la organización. Analiza las consecuencias esperadas e inesperadas de la implementación de un CMS, y hace recomendaciones sobre la gestión del CMS en general.

3.2 Antecedentes Nacionales

De la Cruz (2021), en sus tesis da a conocer sobre el uso de frameworks de desarrollo son cada vez más comunes en equipos de desarrollo de software durante los últimos años, habla también sobre el uso apropiado de un framework específico provee una estructura definida para el proyecto y a la vez mejora notablemente los tiempos de desarrollo y entrega de un producto de software. Sobre la elección de un marco de trabajo suele ser un reto para el equipo de desarrollo a la hora de determinar cuál es el que mejor se acomode a las necesidades del proyecto. Por lo tanto, se realizó la investigación aplicando el diseño cuasi-experimental con enfoque cuantitativo proponiendo de acuerdo al problema cómo elegir el framework que mejore y se adapte a las necesidades de un equipo de desarrollo al momento de implementar una aplicación móvil híbrida que tiene funcionalidades básicas como de (crear, listar, actualizar y eliminar). Al final se concluye que ninguno de los frameworks es mejor que el otro ya que cada uno posee sus atributos particulares o específicos.

Sánchez, Tuesta & Mejía (2015), en este artículo se presenta un análisis comparativo de frameworks utilizando Java para el desarrollo de aplicativos webs, empleando una matriz de evaluación.

Para realizar el analice de estos frameworks seleccionados se efectuó en base a criterios de madurez y documentación, donde se obtuvo como resultado los Frameworks Spring, Struts, JSF, Angular JS para su evaluación. De esta manera, el análisis teórico de frameworks se formuló un método QSOS, creado para definir características comparativas del software de código libre. Por ende, se elaboró un caso práctico de un sistema de votación para conocer los framework evaluados en su implementación y codificación en la práctica. Al final se obtuvo una matriz para la evaluación comparativa para los Frameworks elegidos. Los resultados obtenidos mostraron tanto en funcionalidad como en fiabilidad, mantenibilidad, rendimiento, usabilidad, compatibilidad y portabilidad de cada framework para el desarrollo de aplicaciones web.

Caro (2019), en su tesis de investigación propone un modelo de evaluación de CMS para software libre. Para la formulación de este modelo, se utilizó el método IQMC, el cual se enfoca en un modelo mixto, pues tiene sus propios pasos establecidos y representado por la Norma ISO 25000, soportando las siguientes características: funcionalidad, factibilidad, usabilidad, eficiencia, mantenibilidad y portabilidad, métricas y evaluación de calidad de productos software. obteniendo como resultado una matriz donde prevalece los aspectos fundamentales que debe tener los CMS y sirviéndole como base para establecer criterios en la selección. Aplicándolo en sistemas de gestión como Drupal, Ez Publish, Joomla, Liferay, OpenCms y Wordpress, realizando un análisis comparativo técnico y objetivo. Obteniendo el CMS que más se adapta a las necesidades para el caso práctico de la elaboración del sitio web de la institución educativa Exitus.

CAPÍTULO IV

CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

Como resultado del tema realizado y tomando los capítulos desarrollados en este Informe, se pudo obtener las siguientes conclusiones:

- Se dio a conocer la definición básica, características, tipos, ventajas, desventajas y comparación, tanto de un Framework como de un CMS, para la correcta elección al momento de emprender un proyecto de aplicativo web.

- Ambos sistemas de desarrollo tienen ventajas y desventajas, al elegir la opción de CMS, se creará una solución web con menos tiempo y menos inversión, pero estará limitado a las capacidades de este CMS. Mientras, el desarrollo con Framework permitirá crear soluciones que se adapten o ajusten a los requisitos del proyecto, teniendo costos más elevados y tiempo más largo al momento de hacer la entrega.

- Después de dar a conocer la información de un Framework y CMS, aún no existe una opción correcta o incorrecta, como se puedo observar que cualquiera de los dos sistemas tiene sus ventajas y desventajas, algunos escogerán la opción de framework por su funcionalidad, otros por un CMS que sería aún solución más rentable y rápida.

4.2 Recomendaciones

Entre las recomendaciones se pueden dar las siguientes:

- Se puede dar mejoras a este informe, que se puedan aplicar a los dos métodos de desarrollo web, esto se puede realizar basándose en las debilidades mencionadas y a futuras investigaciones que tengan como objetivo optimizar estos métodos para su correcta utilización.

- Se recomienda tocar estos puntos, como son los Framework y CMS, durante la formación académica del estudiante, debido a que son temas muy interesantes y rentables.

- Independientemente del método que se tome en elección para el desarrollo del aplicativo web, se sugiere tener conocimientos básicos en programación, para el desarrollo de su proyecto.

- En cuanto a los CMS se sugiere capacitarse en los que son más usados por las empresas, siendo estas Drupal, WordPress y Joomla, en lo que son Framework se recomienda comenzar con marcos fáciles y accesibles como lo son Laravel o Yii.

BIBLIOGRAFÍA

Bojorque, R. (2008). Sistemas Gestores de Contenido (CMS). La solución ideal en la Web. *Articulo.* Universidad Politécnica Salesiana. Obtenido de http://www.redalyc.org/articulo.oa?id=505554806008

(2016).*Buenas prácticas en el desarrollo de aplicaciones con ZEND .* Obtenido de https://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/261

Carlos, D. I. (2021). *ANÁLISIS COMPARATIVO ENTRE FRAMEWORKS DE DESARROLLO PARA APLICACIONES MÓVILES HÍBRIDAS.* Universida Señor de Sipan, Lima, Peru. Obtenido de https://orcid.org/0000-0002-2680-2955

Caro Rosales, J. J. (2019). Análisis comparativo de los sistemas de gestión de contenidos (CMS), de software libre, para la implementación de sitios web y aplicación al caso práctico para la Institución Educativa Exitu's. *tesis de grado.* Universidad Nacional de Piura, Piura, Peru. Obtenido de http://repositorio.unp.edu.pe/handle/UNP/1588

Cuerda, X. y. (2007). Introducción a los Sistemas de Gestión de Contenidos (CMS) de código abierto. *Mosaic.*

Educacion, P. (2011). *Ingeniería de Software* (NOVENA EDICIÓN, 2011 ed.). México,: Addison Wesley.

Gutierrez, A. F. (2007). *Kumbia PHP Framework: Porque programar debería ser más fácil.* Obtenido de http://www.kumbia.org/LibroDeKumbia.pdf

(2011).*Instalación – Laravel.* Obtenido de https://www.laravel.com/docs

juntadeandalucia. (s.f.). Obtenido de https://www.juntadeandalucia.es/servicios/madeja/contenido/recurso

MARTÍNEZ VILLALOBOS, G., CAMACHO SÁNCHEZ, G. D., & BIANCHA. (2010). Diseño de Framework web para el Desarrollo Dinamico de Aplicaciones. *Articulo.* Universidad Tecnológica de Pereira, Pereira,

Colombia. Obtenido de https://www.redalyc.org/articulo.oa?id=84917316032

Navarrete, E. N. (2017). *Desarrollo de un marco de trabajo (framework) para el desarrollo de aplicaciones web en la Universidad Nacional de Costa Rica.* Séptima Conferencia de Directores de Tecnología de Información. Obtenido de https://documentas.redclara.net/bitstream/10786/1287/1/131-17-4Desarrollo%20de%20un%20marco%20de%20trabajo%20%28framework%29%20para%20el%20desarrollo%20de%20aplicaciones%20web.pdf

Reenskaug, T. (2018). *MVC: XEROX PARC 1978-79.* Obtenido de https://hmong.es/wiki/Trygve_Reenskaug

Sánchez Acosta Carlos, T. M. (2015). ANÁLISIS COMPARATIVO DE FRAMEWORKS PARA EL DESARROLLO DE APLICACIONES WEB EN JAVA. *articulo.* Universidad Señor de Sipan, Chiclayo. Lambayeque. Perú. Obtenido de http://revistas.uss.edu.pe/index.php/ING/article/view/101

Schwarz, M. (2013). *Design and Analysis of Web Application Frameworks.*

Shan, T. H. (2006). Taxonomy of Java Web Application Frameworks. *IEEE International Conference on Oct. 2006 Digital Object Identifier 10.1109/ICEBE.2006.98.*

Susan Goodwin, N. B. (2006). CMS: sistema de gestión de contenidos/estrategias de gestión de cambios. *Articulo.* Texas A&M, Texas, Usa. Obtenido de https://doi.org/10.1108/07378830610652103

Trninic, M. B. (2010). *Evaluation of web application frameworks.*

FORMATO DE AUTORIZACIÓN DE PUBLICACIÓN EN EL REPOSITORIO INSTITUCIONAL UNAM

Formulario de autorización para la publicación electrónica en la página web del Repositorio Institucional de la Universidad Nacional de Moquegua, de conformidad con la Ley N° 30035 Ley que regula el Repositorio Nacional Digital de Ciencia, Tecnología e Innovación de Acceso Abierto y Resolución de Comisión Organizadora N° 041-2022-UNAM que aprueba el Reglamento que regula el Repositorio Institucional Digital de la UNAM.

1. DATOS DEL AUTOR

Nombres y Apellidos	: HENRRY RENZO TICONA CONDORI
DNI/Carnet de Extranjería/Pasaporte N°	: 43362808
Correo electrónico	: henrrytico26@gmail.com
Teléfono/Celular	: 910924800

2. GRADO, TITULO O ESPECIALIZACIÓN (Marque con una X)

Bachiller	Título Profesional	Magister	Doctor	Segunda Especialidad
	X			

3. TIPO DE PUBLICACIÓN (Marque con una X)

a) Articulo ( )
b) Tesis de pregrado ( )
c) Tesis de Maestría ( )
d) Tesis de Doctorado ( )
e) Monografía (X)
f) Reporte de caso científico o hallazgos ( )
g) Libro ( )
h) Parte de un libro ( )
i) Revisión ( )
j) Conferencia ( )
k) Presentación ( )
l) Artículo preliminar ( )
m) Articulo antes de ser publicado ( )
n) Artículo de prensa ( )
o) Documentos internos ( )
p) Imagen ( )
q) Documentos técnicos ( )
r) Ponencias presentadas en eventos científicos (Jornada, Congreso, Simposio Conferencias, etc.) ( )
s) Informes de suficiencia profesional ( )
t) Patentes producidas por académicos/as ( )

4. TIPO DE TRABAJO DE INVESTIGACIÓN (Marque con una X)

a) Tesis ( )
b) Trabajo de Investigación ( )
c) Trabajo de suficiencia profesional ( )
d) Trabajo académico (X)

5. DATOS DE LA PUBLICACION

a) Título: DESARROLLÓ DE APLICATIVOS WEBS, BENEFICIOS Y DIFERENCIAS AL UTILIZAR UN FRAMEWORK O UN CMS (CONTENT MANAGEMENT SYSTEM).
b) Campo del conocimiento OCDE (Indicar el código de clasificación OCDE de la obra: https://catalogos.concytec.gob.pe/vocabulario/ocde_ford.html) 2.02.04
c) Fecha de publicación: 2022

6. DATOS DEL ASESOR:

a) Nombres y Apellidos: MSG. HUGO EULER TITO CHURA
b) DNI/Pasaporte/Cédula: 01315850
c) ORCID: 0000-0001-8269-0655

FIRMA : .....

DOCUMENTO DE IDENTIDAD : 43362808
NOMBRES Y APELLIDOS : HENRRY RENZO TICONA CONDORI

LICENCIAS

Licencia Estándar

Bajo los siguientes términos, autorizo el depósito de mi trabajo de investigación en el Repositorio Institucional Digital de la UNAM.

Con la autorización de depósito da mi trabajo de investigación, otorgo a la Universidad Nacional de Moquegua, una Licencia No Exclusiva para reproducir. distribuir, comunicar al público, transformar (únicamente mediante su traducción a otros idiomas) y poner a disposición del público mi trabajo de investigación (incluido el resumen), en formato físico o digital, en cualquier medio, conocido o por conocerse, a través de los diversos servicios provistos por la Universidad, creados o por crearse, tales como el Repositorio Institucional Digital de la UNAM, Colección de Tesis, entre otros, en el Perú y en el extranjero, por el tiempo y veces que considere necesarias, y libre de remuneraciones.

En virtud de dicha Licencia, la UNIVERSIDAD NACIONAL DE MOQUEGUA, podrá reproducir mi trabajo de investigación en cualquier tipo de soporte y en más de un ejemplar, sin modificar su contenido, solo con los propósitos de seguridad, respaldo y preservación.

Declaro que el trabajo de investigación es una creación de mi autoría y exclusiva titularidad, o coautoría con titularidad compartida, y me encuentro facultado a conceder la presente licencia y, asimismo, garantizo que dicho trabajo de investigación no infringe derechos de autor de terceras personas.

La UNIVERSIDAD NACIONAL DE MOQUEGUA consignara el nombre del/los autor/es del trabajo de investigación, y no le hará ninguna modificación más de la permitida en la presente licencia.

Autorizo el depósito (marque con una X).

(X) **SI autorizo que se deposite inmediatamente.**

FIRMA : .....

DOCUMENTO DE IDENTIDAD : 43362808

NOMBRES Y APELLIDOS : HENRRY RENZO TICONA CONDORI

FECHA : SETIEMBRE 2022.