UNIVERSIDAD MAYOR DE SAN SIMON

FACULTAD POLITECNICA DEL VALLE ALTO

SISTEMA DE MANEJO Y DIFUSIÓN DE INFORMACÍON DE LA FACULTAD POLITÉCNICA DEL VALLE ALTO

PROYECTO DE GRADO – ADSCRIPCIÓN

DESARROLLADO POR:

DORIS MIREYA TERCEROS OVANDO

TUTOR:

LIC. CORINA JUSTINA FLORES VILLARROEL

COCHABAMBA – BOLIVIA

2017

Título del Proyecto:

SISTEMA DE MANEJO Y DIFUCIÓN DE INFORMACIÓN DE LA FACULTAD POLITÉCNICA DEL VALLE ALTO

FACULTAD POLITÉCNICA DEL VALLE ALTO

AUTOR:

DORIS MIREYA TERCEROS OVANDO

DEDICATORIA:

Agradezco a la vida y a Dios, por darme la oportunidad de vivir y por estar en cada paso que doy, por fortalecer mi espíritu y corazón e iluminar mi mente para culminar este proyecto. Así también, agradezco a todas aquellas personas que han sido mi soporte y compañía durante todo el periodo de estudio.

UNIVERSIDAD MAYOR DE SAN SIMON

FACULTAD POLITÉCNICA DEL VALLE ALTO

SISTEMA DE MANEJO Y DIFUSIÓN DE INFORMACÍON DE LA FACULTAD POLITÉCNICA DEL VALLE ALTO

Autor: D. Mireya Terceros Ovando

Tutor: Lic. Corina Justina Flores Villarroel

Revisor: Arq. Huascar Ariel Rivera Castellón

RESUMEN

La Facultad Politécnica del Valle Alto, es una unidad desconcentrada dependiente de la Universidad Mayor de San Simón, esta se encuentra a 50 Km. de Cercado en la provincia Punata.

Actualmente, cuenta con cinco carreras a nivel Técnico Superior, uno a nivel Técnico Medio y dos programas complementarios a nivel licenciatura. Interna a ella, se encuentra a Unidad de Tecnologías de Información Facultativa (UTI), la cual se encarga del manejo de información referente al estatuto docente, estatuto administrativo y estatuto estudiantil. Este sistema de administrativo fue estudiado y analizado, del cual se diagnosticaron ciertos inconvenientes que conllevan a una falta de acceso y difusión de información entre los diferentes estatutos orgánicos de la Facultad Politécnica del Valle Alto.

En este sentido, el presente trabajo nace de la necesidad que enfrentan los diferentes estatutos orgánicos, teniendo como propósito el desarrollo de un sistema de información web para el manejo y difusión de información que permita mejorar eficientemente el desarrollo de las actividades. Para el desarrollo e implementación de este sistema se utilizó tecnologías Web. El método e instrumento de recolección de datos usados fueron la observación directa, la entrevista no estructurada, encuesta y revisión documental. Los resultados obtenidos con el desarrollo de la herramienta fue la accesibilidad a información y reducción de distancias en cuanto al manejo de información, apoyando así, el proceso de difusión de información verídica y confiable.

CAPITULO I

1. INTRODUCCIÓN

Los sistemas de información han ido evolucionando a los largo de los años empleando diferentes tecnologías, lenguajes y entornos hasta encontrar, en la Web, el hábitat adecuado. La World Wide Web emplea un conjunto de recomendaciones, técnicas y tecnologías robustas que ofrecen un gran rendimiento y fiabilidad, así como un mecanismo de navegación muy amigable para la mayoría de los usuarios. Esto favorece los procesos de enseñanza de los sistemas basados en la Web, ya que parte de conceptos y herramientas ampliamente conocidos, como sucede con los navegadores o métodos de búsqueda y navegación.

Sin embargo, el concepto de Desarrollo Web es la conjunción de dos términos, que se usan en este tiempo con bastante frecuencia. Ya que el desarrollo del software en general, cada vez está más orientada a la **web**. En la actualidad casi todas las instituciones públicas y privadas en el país dependen de complejos sistemas informáticos; y con Internet, los sistemas informáticos deben contar con facilidad de acceso para el usuario, es decir, que el usuario pueda acceder desde cualquier dispositivo que esté conectado a la Web, con solo oprimir un botón. Al aumentar nuestra capacidad de obtener, procesar y difundir información cada vez más complejo, crece incluso con mayor celeridad y muchas veces hasta en tiempo real.

La introducción de nuevos dispositivos de computación y la evolución de los mismos nos obligan a desarrollar sistemas cada vez más complejos y acorde a la necesidad de las instituciones y empresas. Los recursos a compartir se centralizan en una maquina denominada **servidor.** Las demás maquinas, denominadas **estaciones de trabajo,** en este caso hacen uso de los recursos del **servidor**. En nuestro caso usaremos un **servidor** en el **web** que es una red mundial de recursos accesibles mediante la red *Internet.* El **web**  funciona mediante tres mecanismos distintos pero interrelacionados; un esquema que permite nombrar un recurso de forma única en todo el mundo denominado *URL*, protocolos que permiten acceder a los recursos en este caso *http* y finalmente el lenguaje *HTML* que permite desarrollar documentos de Hiper Texto gracias a los cuales se pueden enlazar unos documentos con otros, de esta manera tejiendo esa red virtual que es el *web*.

Asimismo, otro factor que mejoro al paso del tiempo es el desarrollo de ingeniería de software. Este último nos ha permitido mejorar el software y desde entonces hemos desarrollado métodos efectivos de especificación, diseño e implementación del software.

En consecuencia, se desarrolló un sistema que difunda y automatice información, este fue desarrollado para la WEB. Además, fusionara la ingeniería de software y las nuevas tendencias en hardware.

Por lo tanto, la información que se piensa difundir mediante la *WEB* debe tener un buen diseño visual, interfaz de usuario bien pensada, navegación primaria desde el primer pantallazo, repetición de la navegación en el pie de la página, contenido significativo, una sólida página, Acerca de, información de contacto, búsqueda, registro, mapa del sitio, separar el diseño del contenido, validación, compatibilidad del navegador, imágenes web optimizadas, estadísticas, seguimiento y análisis. También nuestro sistema tiene que ser robusto, rápido y amigable.

Sin embargo, las necesidades de información proveniente de las organizaciones han sido transformadas por la expansión acelerada de Internet y el uso de la Intranet, llegando a ser requerida por los miembros, dentro y fuera de la organización y compartiendo con cada uno de los integrantes acceso a información completa sin importar el lugar ni la distancia.

Es por esta razón, que las diferentes entidades organizacionales requieren de nuevas formas de articular sus procesos internos y mejorar la calidad de sus servicios, con el objeto de darse a conocer.

El presente trabajo de grado con modalidad adscripción tuvo como objetivo el desarrollo e implementación de un sistema de información web que sirva de apoyo como medio de manejo y difusión de información eficiente, bajo el uso de tecnologías de ingeniería de software.

* 1. DESCRIPCION DE PROBLEMA

La información se define como un conjunto organizado de datos, que constituyen un mensaje sobre un ente o fenómeno. Durante años, el ser humano ha utilizado diferentes métodos de manejo de la información y toma de decisiones, utilizando como base el papel. Elaborar manualmente los procesos cotidianos de una organización ha sido la forma más común para la realización de tareas.

Actualmente, las organizaciones e incluso de países enteros está basada en la información, esto obliga a las instituciones y usuarios a interactuar directa o indirectamente con sistemas de diferente índole: manuales o automatizados que permitan manejarla de la forma más eficaz. Esto se evidencia en el creciente interés de la población por acercarse a la tecnología en los diferentes sectores de la vida pública.

Es importante mencionar, la integración de particulares e instituciones al trabajo con equipos de cómputo, los cuales agilizan el procesamiento de grandes cantidades de datos y le suministran al usuario la información. Esta integración ayuda a mejorar los procesos administrativos de la organización, lo cual se traduce en mayor competitividad y éxito en las actividades emprendidas.

En ese sentido, la modernización de las organizaciones implica la transformación de la gestión administrativa así como los cambios en las relaciones internas y con el entorno. Por lo que se hace necesario flexibilizar las estructuras hacerlas más livianas y promover esquemas que aseguren mayor compromiso de todas las dependencias de la organización, agilizando los procesos internos.

Es así, como la Facultad Politécnica del Valle Alto unidad dependiente de la Universidad Mayor de San Simón patrocino un estudio en que todos concordaban en que los sistemas de información son necesarios para darse a conocer en la región. Durante un año y con la participación de las distintas autoridades, se buscó un modelo para medir la actuación de las mismas.

Con la experiencia y conocimientos aportados por los participantes se llegó a un producto final: El Sistema de Manejo y Difusión de Información de la Facultad Politécnica del Valle Alto (FPVA), este sistema nos permite difundir información referente a la FPVA, la cual debe ser accesible, actual y fácilmente modificable.

En la actualidad, el Politécnico no cuenta con un software que permita, tanto a estudiantes como docentes acceder a información y menos a difundirla. Para ello, el Politécnico requiere de un sistema que administre adecuadamente toda su información de forma centralizada y eficiente.

Asimismo, la Facultad cuenta con seis carreras; cinco a nivel técnico superior y uno técnico medio, un programa complementario y un programa desconcentrado.

Actualmente, el Politécnico cuenta con una vitrinas donde publican todo tipo de información (anuncios, comunicados, publicaciones, fechas de inscripciones, etc.) referente a la facultad.

Es por ello que en los últimos años las diferentes unidades dependientes de la Universidad Mayor de San Simón han incorporado nuevos sistemas de información que permiten la automatización en la ejecución de los procesos con el fin de aumentar la cantidad y eficacia en la gestión de los servicios que realizan. Estos sistemas denominados flujos de trabajo son sistemas informáticos que permiten la integración de los distintos procesos así como, el control automático de los elementos de esos procesos, desde personas, ordenadores hasta información y documentación. La aplicación de estos nuevos sistemas de gestión a los procesos de difusión de información permite acceder a información y documentos, obteniéndose una mayor eficiencia en el servicio prestado.

La FPVA, objeto de este estudio, cuenta con 62 docentes, 12 administrativos y 1400 alumnos, prestando sus servicios hace 25 años, periodo en el cual se han observado muchas dificultades, por el hecho de que la FPVA se encuentra a 45 Km. de la ciudad y que los diferentes problemas sociales de nuestro país muchas veces han dificultado la llegada del estatuto docente a su fuente de trabajo y al no contar con un medio de información eficiente, el estatuto estudiantil quedaba totalmente desinformado.

En relación a lo anteriormente expuesto, la FPVA, ha considerado el desarrollo de un sistema de difusión de información, como herramienta capaz de manejar información fiable y verídica.

Esta información es utilizada como instrumento eficaz para la administración por lo cual se busca la organización de información haciéndolos de fácil acceso, procesamiento, recuperación y comunicación de acuerdo a los requerimientos de los usuarios involucrados contando así con un sistema funcional que satisfaga sus necesidades. En tal sentido y en función a esta situación se ha planteado el siguiente problema de investigación:

FORMULACIÓN DEL PROBLEMA

¿Cómo desarrollar un sistema automatizado para manejo y difusión de información de la Facultad Politécnica del Valle Alto?

* 1. OBJETIVOS
     1. OBJETIVO GENERAL

Desarrollar un sistema web que difunda información e interactúe con los distintos tipos de usuario, aplicando tecnología móvil, para asegurar el fácil acceso a los usuarios del sistema y de esta forma adecuarse a nuevas tendencias tecnológicas.

* + 1. OBJETIVOS ESPECIFICOS

El sistema web cumplirá con los siguientes objetivos específicos:

1. Mantener informada a la comunidad docente – estudiantil por medio de la implementación de una página web.
2. Informar al plantel docente - estudiantil por carrera con el desarrollo de un portal propio.
3. Generar publicaciones científicas para la comunidad docente – estudiantil por carrera con la implementación de foros de discusión.
4. Publicar información para la comunidad docente – estudiantil mediante el desarrollo de un portal web de difusión de contenido.
5. Fusionar al Politécnico por medio de enlaces con otros portales web como el WEBSISS.
   1. JUSTIFICACIÓN

La importancia que tiene este tipo de sistemas, quedan establecidas por la necesidad de facilitar a los usuarios el acceso, difusión y búsqueda de información, ya que en la actualidad todo tiende a estar interconectado.

* + 1. TECNOLÓGICO

El sistema web es un medio virtual que interrelaciona a un grupo de usuarios, facilitando la difusión y búsqueda de información, con el sistema el usuario únicamente necesitara estar conectado a Internet mediante un dispositivo móvil.

* + 1. SOCIAL

El sistema busca facilitar a los distintos usuarios el acceso y difusión de información, pudiendo hacerlo de cualquier parte del mundo.

* + 1. ECONOMICO

Antes, sin la existencia del Web era muy costosa y tediosa la comunicación entre personas, pero desde la aparición del Internet y la introducción de nuevos dispositivos de computación, el desarrollo web facilitó a los usuarios, siendo esta más accesible, confiable y sobretodo económica. El sistema facilitará la comunicación entre los usuarios de ahí en adelante.

* 1. LIMITES Y ALCANCES

Para el desarrollo e implementación del sistema se observaron los siguientes aspectos:

ALCANCES

* Una vez terminado el sistema web para el Politécnico, este será lanzado en la web, siempre y cuando se cuente con un IP público, un servidor y un dominio.
* Los logotipos de cada carrera serán proveídos por la facultad.

LIMITES

* El Politécnico deberá tener sumo cuidado en la difusión de información.
* El Politécnico deberá hacer el uso adecuado del sistema web.
* El Politécnico tendrá la obligación de asignar un administrador del sistema que tenga la responsabilidad de administrarla de forma correcta.

CAPITULO II

1. MARCO TEORICO
   1. INTRODUCCIÓN

En relación al marco teórico, Carlos Sabino afirma que “el planteamiento de una investigación no puede realizarse si no se hace explicito aquello que nos proponemos conocer: es siempre necesario distinguir entre lo que se sabe y lo que no se sabe con respecto a un tema para definir claramente el problema que se va a investigar”. El correcto planteamiento de un problema de investigación nos permite definir sus objetivos generales y específicos, como así también la delimitación del objeto de estudio.

El autor agrega que ningún hecho o fenómeno de la realidad puede abordarse sin una adecuada conceptualización. El investigador que se plantea un problema, no lo hace en el vacío, como si no tuviese la menor idea del mismo, sino que siempre parte de algunas ideas o informaciones previas, de algunos referentes teóricos y conceptuales, por más que esto no tenga todavía un carácter preciso y sistemático.

El marco teórico, tiene el propósito de dar a la investigación un sistema coordinado y coherente de conceptos y proposiciones que permitan abordar el problema. “Se trata de integrar al problema dentro de un ámbito donde este cobre sentido, incorporando los conocimientos previos relativos al mismo y ordenándolos de modo tal que resulten útil a nuestra tarea”.

* 1. DESCRIPCIÓN DE LA INSTITUCIÓN

La Facultad Politécnica del Valle Alto (FPVA), es una unidad desconcentrada dependiente de la Universidad Mayor de San Simón (UMSS). La FPVA viene consolidándose desde años atrás en la región del Valle Alto y es una unidad facultativa emergente que forma profesionales con valores éticos, morales, conocimientos sólidos, habilidades, destrezas enmarcados en la teoría y la práctica aportando a la educación e innovación tecnológica, para el desarrollo de la región y del país mejorando la calidad de vida de sus habitantes. Esta institución está dirigida por diferentes espectros sociales: el estatuto docente, el estatuto administrativo y el estatuto estudiantil, aportando a cada uno de ellos la solución más idónea para sus necesidades en cada momento. Cuenta para ello con un equipo humano con formación personalizada en la propia entidad, en sus diferentes áreas y departamentos.

Misión

La FPVA como Unidad Académica de la UMSS, afirma el liderazgo regional en especial en el Valle Alto como Institución de Educación Superior en el cumplimiento de formar recursos humanos de carácter técnico, capaces de transformar la realidad mejorando los procesos de recolección y transformación de recursos, contribuyendo al Valle Alto en su trabajo de lograr un desarrollo sostenible, mejorando la calidad de vida de sus habitantes y tomando en cuenta a los sectores más desfavorecidos a través de sus tres líneas estratégicas.

Visión

Formar profesionales de excelencia capaces de transformar la realidad regional mejorando los procesos de producción, dando énfasis al desarrollo sostenible mediante las ciencias medio ambientalistas y adecuando sus principios, fines y objetivos a los modernos paradigmas sociales.

Optimizar el uso de sus RRHH y dar la flexibilidad necesaria para adaptarse a las necesidades regionales y del Valle Alto.

Contar con profesionales necesarios e idóneos, capaces de impartir los diversos cursos de especialización.

Relacionarse con la comunidad, realizando trabajos con las instituciones regionales, tanto municipales como de desarrollo, en el campo de asesoramiento, diseño y participación comunitaria.

Estructura Orgánica

* 1. ANTECEDENTES DE LA INVESTIGACIÓN

Los antecedentes constituyen referencias investigativas efectuadas anteriormente, cuyos temas y resultados consolidan el estudio actual. En este sentido, se exponen estudios que refieren parte de los criterios analizados en relación al desarrollo de un sistema automatizado para la difusión de información.

CAPITULO III

1. ÁREA DE APLICACIÓN

CAPITULO IV

1. METODOLOGÍA
   1. INTRODUCCIÓN

Para llevar a cabo el desarrollo de un sistema de software es importante considerar el estudio de metodologías y técnicas de programación que nos colaboren en el desarrollo de software.

En este sentido el Método de la 6’D es un método de tipo estructurado fácil de comprender y guía paso a paso hasta la solución del problema. Está compuesto de seis etapas, cada una de las cuales consta de una serie de pasos que se van modificando o ajustando, dependiendo del grado de complejidad del problema y las herramientas que se utilicen para su solución.

* 1. MARCO DE TRABAJO 6’D

En el método de las 6’D la solución de los problemas comienza con el enunciado del problema y luego se desarrolla en cada una de las etapas consecutivas, hasta llegar a la etapa de documentación.

Con la finalidad de recordar el método, se ha establecido que el nombre de cada una de sus seis etapas comience con la letra “D”. (Figura 4.1 Método de las 6’D)



El **Método de las 6’D,** soluciona problemas sencillos pero es necesario conocer los conceptos fundamentales de modelamiento, algoritmo y programación, y dominar el uso de estructuras lógicas, instrucciones o sentencias de bifurcación y las estructuras de datos conocidas como variables.

Así también, es posible plantear soluciones más complejas con el **Método de las 6’D**, pero para lo cual se deben dominar conceptos como Objetos, Métodos, Encapsulamiento, Herencia, Polimorfismo, Interfaces Graficas, Conexión a bases de datos, entre otros.

También es importante mencionar que, dependiendo de la complejidad del problema, los pasos especificados en cada una de las etapas del método se puede redefinir o modificar.

* + 1. ETAPAS
       1. ETAPA 01: DESCRIPCIÓN DEL PROBLEMA (2 horas máximo)

En esta primera etapa identificamos cual es el problema que se desea resolver, dividiéndolos en módulos y tareas. Al parecer, es algo sumamente sencillo, pero muchas veces resulta una tarea agotadora, ya que existen muchas opiniones acerca de cuál es el problema central. Identificarlo es una de las tareas más importantes.

Una vez identificado el problema, es necesario comprenderlo en su totalidad, es decir, entender que es exactamente lo que se desea resolver. Finalmente, se debe escribir un enunciado claro, concreto y conciso del problema a resolver.

* + - 1. ETAPA 02: DEFINICIÓN DE LA SOLUCIÓN (2 horas máximo)

En la etapa 02, es necesario estudiar a fondo el problema para poder solucionarlo, saber exactamente en que consiste y descomponerlo en cada una de sus partes, facilitando su comprensión y posterior solución. Esta es una regla que siempre debe ser aplicada.

Una vez entendido el problema se está en condiciones de estudiarlo a fondo y plantear diversas alternativas que permitan solucionarlo, aplicando siempre la más adecuada.

* + - 1. ETAPA 03: DISEÑO DE LA SOLUCIÓN (2 horas máximo)

Una vez definida la solución se procederá a diseñar la lógica, modelando y desarrollando algoritmos.

Para el modelado de la solución del problema se utilizara el Lenguaje Unificado de Modelado (UML), que es una herramienta utilizada para describir clases, objetos y sus relaciones.

Para el desarrollo de algoritmos se utilizaran Pseudocódigos y Diagramas de Flujo (DF), que son herramientas utilizadas para diseñar algoritmos de los diferentes métodos de una clase.

Para finalizar esta etapa es necesario verificar si se han incluido soluciones para todas las formas en que se presenta el problema, para ello utilizaremos “Pruebas de escritorio”.

* + - 1. ETAPA 04: DESARROLLO DE LA SOLUCIÓN

Para la etapa 04, una vez previstas todas las posibilidades y alternativas que puedan presentarse y que pasen sin inconvenientes por la clase y los algoritmos, se procederá a la codificación del problema.

La codificación involucra traducir los diagramas, las especificaciones de las clases y los pasos del algoritmo de cada método, en sentencias de un lenguaje de programación determinado. Estas sentencias son almacenadas en un proyecto lógico, y constituyen lo que la computadora podrá ejecutar.

* + - 1. ETAPA 05: DEPURACIÓN Y PRUEBAS

Luego de codificar el modulo y/o clases, deberán ser probados mediante su ejecución. Al realizar este paso puede surgir diferentes tipos de errores, siendo lógicos o de sintaxis.

Para lo cual hay que corregir, anular, modificar o crear nuevas sentencias, volver a probar el sistema y continuar con la corrección y pruebas hasta obtener la respuesta deseada.

* + - 1. ETAPA 06: DOCUMENTACIÓN

Finalmente, para esta etapa se recompila toda la documentación generada en las etapas anteriores, la cual será de utilidad para la elaboración del manual técnico. El tiempo dedicado a esta etapa será de mucha ayuda para desarrollar buenos hábitos para desarrollo de software a nivel profesional.

La **Ingeniería Reversa** nos permite crear o actualizar el modelo preliminar de Diseño de la solución (Etapa 03), para lo cual es posible programar algunas partes que no estén especificadas en la Etapa 03. A partir de esta solución, se actualizan los diagramas hasta llegar a la solución deseada.

* 1. MODELAMIENTO

El modelamiento es la forma en la que se representa la solución de un problema del mundo real en términos de un modelo. Esta es una representación gráfica o simbólica de algún aspecto del mundo que se encuentra bajo observación o estudio. En este caso, como ya lo mencionamos haremos uso de UML.

Para desarrollar un modelo y representarlo en UML es necesario conocer todos los conceptos relacionados con el desarrollo de software orientado a objetos.

* + 1. PAQUETE

Los paquetes nos permiten organizar las clases de un modelo. Un paquete contiene clases con funciones similares. (Figura 4.2 Paquete)



* + 1. CLASE

La clase es un modelo que se utiliza para describir uno o más objetos del mismo tipo. La forma más sencilla de definir una clase, dícese que es un conjunto de atributos y métodos; es una abstracción y no representa a un objeto en particular. Con el método UML, representaremos una clase como se muestra a continuación: (Figura 4.3 Modelo UML Representación de la Clase)



* + 1. OBJETO

Un objeto es una unidad dentro de un programa o sistema que consta de un estado y de un comportamiento, que a su vez constan respectivamente de datos almacenados y de tareas realizables durante el tiempo de ejecución. Un objeto se crea o instancia a partir de una clase. (Figura 4.4 Modelo UML 2.0 Representación del Objeto)



* + 1. DIAGRAMA DE PAQUETES

El diagrama de paquetes nos permite especificar y visualizar las relaciones de dependencia existentes entre los paquetes que forman parte de una solución. Una relación de dependencia entre dos o más paquetes se establece cuando las clases que pertenecen a un paquete pueden tener acceso a todas o algunas de las clases que pertenecen a otro paquete. (Figura 4.5 Diagrama de Paquetes)



* 1. EL PROCESO

En el método de las 6’D, el proyecto a desarrollar se ejecuta por la recolección de datos en bloques cortos y fijos, este se inicia en la etapa 01 Descripción del problema tomando un lapso aproximado de un mes, cada iteración tiene que proporcionar un resultado completo por modulo, la culminación de cada módulo será un incremento al producto final para la entrega al cliente cuando este lo solicite.

El proceso parte de la lista de requerimientos priorizada del producto, que actúa como plan de proyecto. Por lo cual el cliente priorizara los módulos como objetivos balanceando el valor que le aportan y quedan repartidos en iteraciones. De manera regular el cliente puede maximizar la utilidad de lo desarrollado y el retorno de inversión mediante la **ingeniería reversa** de los objetivos que realiza la etapa 05: Depuración de pruebas, es decir, que lo desarrollado si no llegara a mostrar el resultado esperado, se debe replantear el problema con la etapa 03: Diseño Solución, para que de esta manera se llegue a los objetivos planteados y a la satisfacción del cliente.

CAPITULO V

1. FRAMEWORK LARAVEL 5.4
   1. INTRODUCCION

Laravel en sus versión 5.4 es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP5, siguiendo el patrón arquitectónico Modelo Vista Controlador (MVC). También Laravel es un sistema de empaquetado modular con un administrador de dependencias dedicado, diferentes formas de acceder a bases de datos relacionales, utilidades que ayudan en el despliegue y mantenimiento de aplicaciones y una orientación ASP. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el “código espagueti”.

Fue creada por Taylor Otwell y puso a disposición el 9 de junio de 2011. Pero a partir de marzo de 2015, Laravel es considerado como uno de los frameworks PHP más populares.

* 1. HISTORIA

Taylor Otwell creó Laravel como un intento de proporcionar una alternativa más avanzada al framework CodeIgniter, que no proporcionaba ciertas características tales como soporte incorporado para la autenticación y autorización del usuario. La primera versión beta de Laravel se puso a disposición el 9 de junio de 2011, seguida por la versión de Laravel 1 lanzada el mismo mes. Laravel 1 tiene un soporte integrado para autenticación, localización, modelos, vistas, sesiones, enrutamiento y otros mecanismos, pero carecía de soporte para los controladores que le impidieron ser un verdadero framework MVC.

Laravel 2 fue lanzado en septiembre de 2011, trayendo varias mejoras del autor y la comunidad. Las nuevas características principales incluyeron el soporte para los controladores, lo que hizo que Laravel 2 fuera un framework totalmente compatible con MVC, soporte incorporado para soportar el principio de control de inversión (loC) y un sistema de plantillas llamado Blade. Como un inconveniente, el soporte para paquetes se eliminó.

Laravel 3 fue lanzado en febrero del 2012 con un conjunto de nuevas características incluyendo la Interfaz de Línea de Comandos (CLI) llamada Artisan, soporte integrado para Sistemas de Gestión de Bases de Datos, la migración e bases de datos como una forma de control de versiones para bases de datos, eventos, y un sistema de empaquetado llamado Bundles. El aumento de usuarios y popularidad de Laravel promovió el lanzamiento de Laravel 3.

Laravel 4, con Illuminate, fue lanzado en mayo de 2013. Se reescribió por completo el framework Larave, la migración de su diseño en un conjunto de paquetes distribuidos a través de Composer, que sirve como un gestor de paquetes a nivel de aplicación. Esto mejoro la extensibilidad de Laravel 4, que fue lanzada a la par de su lanzamiento oficial. Otras nuevas funciones de la versión4 de Laravel incluyen la preselección de la base de datos, el soporte de mensajes queues, soporte de envío de diferentes tipos de correos electrónicos y soporte en la eliminación de registros de la base de datos llamada eliminación suave.

Laravel 5 fue lanzado en febrero de 2015 como resultado de cambios internos. Las nuevas características del lanzamiento de Laravel 5 incluyen soporte para la programación de tareas ejecutadas periódicamente a través del paquete llamado Scheuduler, una capa de abstracción llamada Flysystem que permite el almacenamiento remoto para ser usado de la misma manera que los sistemas de archivos locales, mejorando la gestión de paquetes activos a través de Elixir, simplificado el manejo de autentificación externa por medio del paquete Socialite. Laravel 5 tambien introdujo una nueva estructura de directorios para aplicaciones desarrolladas.

Laravel 5.1, lanzado en junio de 2015, en esta ocasión es el primer lanzamiento de Laravel para recibir soporte a largo plazo (LTS), con disponibilidad prevista para correcciones de errores por tres años y parches de seguridad.

Laravel 5.3, lanzado el 23 de agosto de 2016, entre sus nuevas características se centran en mejorar la velocidad del desarrollador añadiendo mejoras adicionales en las tareas comunes.

Laravel 5.4, lanzado el 24 de enero de 2017. Esta versión tiene nuevas características como Laravel Dusk, Laravel Mix, Blade Components y Slots, Markdown Emails, Automatic Facades, Route Improvements, Higher Order Messaging for Collections, entre otros.

* 1. CARACTERICTICAS DE LARAVEL 5.4

Las siguientes características sirven como puntos clave de diseño de Laravel:

* Los paquetes proporcionan un sistema de empaquetado modular, con las características incluidas para la adición fácil a las aplicaciones. Ademas, Laravel utiliza Composer como un gestor de dependencias para agregar paquetes PHP específicos de Laravel y disponibles para el repositorio Packagist.
* ORM Eloquente (Modelo Objeto-Relación) es una implementación avanzada de PHP del patrón de registro activo, proporcionando al mismo tiempo métodos internos para imponer restricciones en las relaciones entre objetos de la base de datos. ORM Eloquente presenta tablas de bases de datos como clases, con las instancias de objeto unidas a las filas de la tabla como única.
* Query Builder, proporcióna una alternativa de acceso a bases de datos más directa al ORM Eloquente. En lugar de requerir que las consultas SQL se escriban directamente, el constructor de consultas de Laravel proporciona un conjunto de clases y métodos capaces de generar consultas de forma programática. También permite el almacenamiento en cache seleccionable de los resultados de la consultas ejecutadas.
* La lógica de la aplicación es una parte integral de las aplicaciones desarrolladas, implementadas mediante controladores o como parte de las declaraciones de ruta. La sintaxis utilizada para definir la lógica de la aplicación es similar a la utilizada por Sinatra framework.
* El enrutamiento inverso define una relación entre los enlaces y las rutas, lo que nos hace posible que los cambios posteriores en las rutas se propaguen automáticamente en enlaces relevantes. Cuando los vínculos se crean mediante el uso de nombres de rutas existentes, los apropiados identificadores de recursos uniformes (URIs) son creados por Laravel.
* Restful Controllers proporcionan una forma opcional de separar la lógica de servicios requeridos HTTP GET and POST.
* Class Auto Loading Providers carga de forma automática las clases PHP sin necesidad de mantenimiento manual de inclusión de rutas. On-demand loading prevents impide la inclusión de componentes innecesarios, por lo que solo se cargan los componentes realmente necesarios.
* View composers serve son unidades de código lógico que pueden ser ejecutadas cuando se carga una vista.
* Blade es una plantilla ingeniosa que combina uno o más plantillas con un modelo de datos para obtener como resultado vistas, haciendo que transmita las plantillas en código PHP para mejorar el rendimiento. Blade también proporciona un conjunto de sus propias estructuras de control tales como declaración de condicionales y bucles, que se asignan internamente a PHP. Además, los servicios de Laravel pueden ser llamados a partir de plantillas Blade, y su motor de plantillas se puede ampliar con directivas personalizadas.
* IoC containers permiten generar objetos nuevos siguiendo el principio de control de inversión (IoC), en el que el framework llama al código de aplicación o tarea específica, con la instanciación y referencia opcional de objetos nuevos.
* Migrations provide proporciona un sistema de control de versiones para el esquema de la base de datos, lo que permite asociar cambios en la base del código de la aplicación y los cambios necesarios en el diseño de la bases de datos. Como resultado, la característica simplifica el despliegue y la actualización de las aplicaciones basadas en Laravel.
* Database seeding provides proporciona una forma de rellenar las tablas de bases de datos con los datos predeterminados que se pueden utilizar para las pruebas de aplicación que se realizaran como parte de la configuración inicial de la aplicación.
* Unit testting proporciona como parte integral de Laravel, que contiene pruebas unitarias que detectan y previenen regresiones en el framework. Las pruebas de unidad se pueden ejecutar a través de la línea de comando de artisan provider.
* Automatic pagination simplifica la tarea de implementar la paginación, reemplazando métodos habituales de implementación manual con métodos automatizados integrados a Laravel.
* Form Request es una característica de Laravel, que sirve como base para la validación de la entrada de formularios mediante los eventos de listeners, dando como resultado la invocación automática de métodos de validación de formularios y la generación de un formulario actual.
* Homestead – Vagrant Virtual Machine que proporciona a los desarrolladores de Laravel todas las herramientas necesarias para desarrollar Laravel.
  1. ARTISAN CLI

Laravel’s command-line interface (CLI), llamado Artisan, fue introducida en la Laravel 3 con un conjunto limitado de capacidades. La posterior migración de Laravel en una arquitectura basada en Composer permitió a Artisan incorporar diferentes componentes del framework Symfony, resultado es la disponibilidad de características adicionales de Artisan en Laravel 4.

Las características de Artisan se asignan a diferentes subcomandos de la utilidad artisan command-line, proporcionando funcionalidad que ayuda en la gestión y la construcción de aplicaciones basadas en Laravel. Los usos más comunes de Artisan incluyen la gestión de migraciones de bases de datos, generando código estandarizado para nuevos controllers y migrations; Este último libera al desarrollador de crear estructura de código adecuado. La funcionalidad y las capacidades de Artisan también se pueden aplicar mediante la implantación de nuevos comandos personalizados, que puedes utilizar para automatizar las tareas recurrentes específicas de la aplicación.