

Universidad de las Ciencias Informáticas

Facultad 4



**Título: Procedimiento de trabajo para el
Desarrollo de Portales Web en el CMS Drupal**

Trabajo de Diploma para optar por el título de
Ingeniero Informático

Autor: Alejandro Javier Marrero Alamo

Tutor: Ing. Yanay Viera Lorenzo

La Habana, Junio 2009

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo la facultad 4 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Autor

Alejandro Javier Marrero Álamo

Tutor

Yanay Viera Lorenzo

DATOS DE CONTACTO

"[insertar breve curriculum e información de contacto del tutor]"

"[insertar breve curriculum e información de contacto del asesor]"

"[insertar breve curriculum e información de contacto del consultante]"

AGRADECIMIENTOS

"[Insertar agradecimientos (opcional)]"

DEDICATORIA

"[Insertar dedicatoria (opcional)]"

RESUMEN

Hoy en día; instituciones, empresas e incluso grupos de personas tienen la creciente necesidad de exponer a usuarios el acceso a recursos y servicios a través de la Web. En aras de suplir esta necesidad la industria del software concibe el desarrollo de Portales Web que satisfacen la demanda de propiciar, entre otros, servicios informativos, buscadores, foros y transacciones electrónicas fundamentalmente.

El desarrollo de un Portal Web requiere una serie de personas involucradas, de actividades y de artefactos que favorezcan un producto final adecuado y con la calidad requerida. Para aliviar el contenido de trabajo para Portales Web, y de manera general en sitios Web, se emplean los Sistemas de Gestión de Contenidos.

El propósito de la presente investigación es establecer un procedimiento de trabajo que favorezca el desarrollo de Portales Web basados en un CMS en específico, Drupal. Para cumplimentar este objetivo se definen los artefactos para una buena y completa documentación, se describen las fases y actividades y se establecen los roles que juegan papel en el desarrollo de un Portal Web basado en Drupal.

Como producto de la investigación se persigue que exista un trabajo, por parte del equipo de desarrollo, más organizado, eficiente, fluido y en el menor tiempo posible.

PALABRAS CLAVE

actividad, artefacto, fase, Portal Web, rol, Sistema de Gestión de Contenidos.

TABLA DE CONTENIDOS

INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	4
1.1 Introducción	4
1.2 Sistemas de Gestión de Contenidos	4
1.2.1 ¿Qué es un CMS?	5
1.2.2 Necesidad de un CMS.....	7
1.2.3 Presente y futuro de los CMS	8
1.2.4 CMS Drupal.....	9
1.3 Metodologías de Desarrollo	11
1.3.1 Web Site Design Method (WSDM).....	12
1.3.2 UML-Based Web Engineering (UWE).....	14
1.3.3 Navigational Development Techniques (NDT)	15
1.3.4 Web Modeling Language (WebML).....	16
1.3.5 Rational Unified Process (RUP).....	18
1.3.6 eXtreme Programming (XP)	20
1.4 Análisis de las Metodologías Anteriores	23
1.5 Conclusiones	26
CAPÍTULO 2: PROCEDIMIENTO DE TRABAJO	27
2.1 Introducción	27
2.2 Roles involucrados en el procedimiento de trabajo	27
2.3 Fases del procedimiento de trabajo	28
2.3.1 Fase 1: Inicio.....	29
2.3.2 Fase 2: Definiciones para el desarrollo	34
2.3.3 Fase 3: Diseño, implementación y pruebas.....	38
2.3.4 Fase 4: Implantación y publicación	44
2.4 Conclusiones	47
CAPÍTULO 3: VALIDACIÓN DEL PROCEDIMIENTO PROPUESTO	48
CONCLUSIONES	58

RECOMENDACIONES	59
BIBLIOGRAFÍA	60
GLOSARIO DE TÉRMINOS	62
ANEXOS	64

INTRODUCCIÓN

En la actualidad, el mundo está inmerso en una sociedad donde la información, las comunicaciones y la tecnología tienen un papel importante en la vida cotidiana. Han surgido nuevos medios para difundir y compartir la información y la manera de gestionarla es una estrategia más de una organización en desarrollo. La necesidad de gestionar la información haciéndola accesible y de forma ágil para la audiencia ha hecho que instituciones, personas y empresas, de manera general, tomen como vía estratégica el posicionamiento de un sitio Web que les conceda a los usuarios la información que necesitan. Debido a esto millones de usuarios se conectan a Internet para buscar o acceder a información. La mayoría de los sitios que se pueden encontrar en Internet o bien están orientados a satisfacer las necesidades potenciales del cliente o reflejan en ellos la realidad de alguna organización, así como sus ventajas comparativas o competitivas.

Los Portales Web son una solución factible a la necesidad de proporcionar servicios informativos e interactivos de manera instantánea, como lo exige la era actual. Estos servicios incluyen foros, noticias, encuestas, buscadores, transacciones electrónicas o de manera general información en distintos formatos. Si bien al comienzo los Portales Web eran sólo modelos basados en páginas estáticas, hoy en día pueden llegar a ser aplicaciones Web muy complejas que gestionan gran cantidad de información y brindan muchas funcionalidades a los usuarios. Debido a la complejidad que iban alcanzando los Portales Web, fueron originalmente creados los Sistemas de Gestión de Contenidos (CMS), permitiendo que fuese más fácil crear, mantener, administrar y publicar sitios Web. Drupal es un típico CMS que permite gestionar fácilmente Portales Web.

“Drupal es un sistema de gestión de contenidos de código abierto. Es multiusuario, multiplataforma, multilinguaje, extensible, modular.” (Martin y otros, 2006)

Los desarrolladores de Portales Web a nivel mundial comienzan a utilizar los CMS con más frecuencia, debido a que suponen un ahorro considerable y acortan el tiempo de realización de un sitio Web. La Universidad de las Ciencias Informáticas (UCI), como pilar en la construcción de software del país, ha reconocido la importancia de usar los CMS y en la actualidad se emplean en varios proyectos. El CMS Drupal ha sido utilizado en varios proyectos, teniendo gran aprobación dentro de la comunidad universitaria. Aunque muchos equipos de desarrollo ven en el uso de Drupal la vía correcta para desarrollar sus Portales Web, estos encuentran varios obstáculos debido principalmente a la falta de experiencia en la Universidad con el uso de los CMS, y la falta de personal capacitado para enseñar a

utilizar estas herramientas. Los equipos de trabajo necesitan invertir parte del tiempo destinado a la realización del proyecto a investigar y a adquirir el conocimiento para poder desarrollar la aplicación Web. Estos equipos de desarrollo tienen dificultad en definir un proceso de desarrollo que tienen que seguir para elaborar el sitio Web, por lo que incurren generalmente en los siguientes errores:

- Atraso en el tiempo planificado en el cronograma del proyecto: A menudo los desarrolladores deben parar su trabajo debido a que no tienen un procedimiento a seguir y deben evaluar constantemente si el trabajo realizado hasta el momento es correcto.
- Incorrecta planificación de las actividades del proyecto: Los líderes de proyectos no conocen la secuencia correcta de tareas que conduzcan al equipo de desarrollo a completar el proyecto de una manera rápida y eficaz.
- Aumento del costo del proyecto: Los errores que se pueden llegar a cometer debido a la ausencia de un procedimiento de trabajo unido al aumento del tiempo de realización, conllevan gastos que no fueron planificados inicialmente dentro del presupuesto destinado al proyecto.
- Deterioro de la calidad del producto final: Todos los factores anteriormente mencionados atentan contra la calidad del producto que se desea realizar.
- Insatisfacción con el cliente: El cliente no queda conforme, pues surgen varios problemas que no estaban previstos en el acuerdo inicial. Esto influye en que los clientes pierdan confianza en la empresa productora y disminuyan, para esta, los contratos de trabajo.

Analizando todos estos inconvenientes se considera que es apremiante la necesidad de definir un procedimiento de trabajo que permita conocer qué roles realizan las actividades en el desarrollo de Portales Web, cuáles artefactos son necesarios y las fases del proceso de desarrollo en las que se deben enmarcar estos elementos. Debido a esto se impone el siguiente **problema**: ¿Cómo llevar a cabo el procedimiento de trabajo para el desarrollo de Portales Web basados en el CMS Drupal. Este problema está enmarcado en el **objeto de estudio**: las metodologías de desarrollo, y como **campo de acción** se define: los procedimientos de trabajo para el desarrollo de Portales Web utilizando CMS Drupal.

El **objetivo** principal del presente trabajo es crear una propuesta de un procedimiento de trabajo para el desarrollo de Portales Web utilizando el CMS Drupal. Para alcanzar el objetivo propuesto se realizarán las siguientes **tareas**:

- Hacer un análisis de las metodologías de desarrollo que rigen el desarrollo de software en la actualidad y de software para la Web particularmente.

- Hacer un análisis de las características del CMS Drupal y su influencia en el proceso de desarrollo de los Portales Web.
- Definir fases, artefactos, actividades y roles involucrados en el desarrollo de los Portales Web basados en el CMS Drupal.
- Validar la solución propuesta.

La estructuración del contenido queda dividida en 3 capítulos:

- Capítulo 1: Fundamentación Teórica: Se muestran las características de los CMS de manera general, así como se abunda en las características y tendencias del CMS Drupal. Se incluye información sobre las metodologías de desarrollo de software en general y otras que son para realizar Portales Web específicamente; así como un análisis sobre las mismas.
- Capítulo 2: Procedimiento de Trabajo: Se muestran todos los elementos que componen el procedimiento de trabajo. Se definen las cuatro fases de desarrollo que componen el procedimiento, así como las actividades que se realizan en cada fase, los trabajadores y los artefactos que se deben obtener al terminar cada fase de desarrollo.
- Capítulo 3: Validación del Procedimiento de Trabajo: Se muestra el proceso de validación del procedimiento de trabajo propuesto. Se muestran todos los pasos del método escogido para validar la propuesta; desde la selección de los expertos y la conformación del cuestionario hasta los cálculos finales que determinan la validez del procedimiento del trabajo.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

Este capítulo muestra un panorama sobre los Sistemas de Gestión de Contenidos, partiendo de las características más generales a las más específicas, dejando claro su importancia y beneficios para realizar aplicaciones Web y profundiza en Drupal, analizando sus características y tendencias actuales. Además se analizan varias de las metodologías más usadas a nivel mundial para desarrollar software, y algunas que se utilizan para crear aplicaciones Web en específico. Por último se define cuáles de estas metodologías se utilizarán como base de la investigación.

1.2 Sistemas de Gestión de Contenidos

Realizar un sitio Web puede ser un trabajo complicado y muy laborioso si no se dispone de las herramientas adecuadas. En el pasado las herramientas eran básicamente editores que permitían generar una página, que evolucionaron para incorporar el control de la estructura de la Web y otras funcionalidades, pero en general estaban enfocadas más a la creación que al mantenimiento. En los últimos años se ha desarrollado el concepto de Sistema de Gestión de Contenidos (*Content Management Systems* o CMS). Se trata de herramientas que permiten crear y mantener una Web con facilidad, encargándose de los trabajos más tediosos que hasta ahora ocupaban el tiempo de los administradores de las Webs.

Teniendo en cuenta el ahorro que supone la utilización de estas herramientas, y el coste de desarrollarlas, sería lógico esperar que su precio fuera muy elevado. Eso es cierto para algunos productos comerciales, pero existen potentes herramientas de gestión de contenidos de acceso libre, disponibles con licencias de código abierto.

Los gestores de contenidos proporcionan un entorno que posibilita la actualización, mantenimiento y ampliación de la Web con la colaboración de múltiples usuarios. En cualquier entorno virtual ésta es una característica importante, que además puede ayudar a crear una comunidad cohesionada que participe más de forma conjunta.

1.2.1 ¿Qué es un CMS?

El CMS es un software que se utiliza principalmente para facilitar la gestión de aplicaciones Webs, ya sea en Internet o en una Intranet, y por eso también son conocidos como gestores de contenido Web (*Web Content Management* o WCM). Hay que tener en cuenta, sin embargo, que la aplicación de los CMS no se limita sólo a las Webs.

A continuación se muestra una propuesta de las funcionalidades de los CMS dividida en cuatro categorías (Robertson, 2003)

- Creación de contenido

Un CMS aporta herramientas para que los creadores sin conocimientos técnicos en páginas Web puedan concentrarse en el contenido. Lo más habitual es proporcionar un editor de texto WYSIWYG¹, en el que el usuario ve el resultado final mientras escribe, al estilo de los editores comerciales, pero con un rango de formatos de texto limitado. Esta limitación tiene sentido, ya que el objetivo es que el creador pueda poner énfasis en algunos puntos, pero sin modificar mucho el estilo general del sitio Web.

Hay otras herramientas como la edición de los documentos en XML², utilización de aplicaciones ofimáticas con las que se integra el CMS, importación de documentos existentes y editores que permiten añadir marcas, habitualmente HTML³, para indicar el formato y estructura de un documento.

Un CMS puede incorporar una o varias de estas herramientas, pero siempre tendría que proporcionar un editor WYSIWYG por su facilidad de uso y la comodidad de acceso desde cualquier ordenador con un navegador y acceso a Internet.

Para la creación del sitio propiamente dicho, los CMS aportan herramientas para definir la estructura, el formato de las páginas, el aspecto visual, uso de patrones, y un sistema modular que permite incluir funciones no previstas originalmente.

- Gestión de contenido

Los documentos creados se depositan en una base de datos central donde también se guardan el resto de datos de la Web, cómo son los datos relativos a los documentos (versiones hechas, autor, fecha de publicación y caducidad, datos y preferencias de los usuarios, la estructura de la Web).

¹ What You See Is What You Get. Técnica que ofrece la reproducción exacta en pantalla de un texto tal como aparecería después en formato impreso.

² eXtensible Markup Language. Estándar de Internet que permite una estructura lógica y flexible para archivos de documentos.

³ HyperText Markup Language). Lenguaje estándar para describir el contenido y la apariencia de las páginas en el WWW.

La estructura de la Web se puede configurar con una herramienta que, habitualmente, presenta una visión jerárquica del sitio y permite modificaciones. Mediante esta estructura se puede asignar un grupo a cada área, con responsables, editores, autores y usuarios con diferentes permisos. Eso es imprescindible para facilitar el ciclo de trabajo (*workflow*) con un circuito de edición que va desde el autor hasta el responsable final de la publicación. El CMS permite la comunicación entre los miembros del grupo y hace un seguimiento del estado de cada paso del ciclo de trabajo.

- **Publicación**

Una página aprobada se publica automáticamente cuando llega la fecha de publicación, y cuando caduca se archiva para futuras referencias. En su publicación se aplica el patrón definido para toda la Web o para la sección concreta donde está situada, de forma que el resultado final es un sitio Web con un aspecto consistente en todas sus páginas. Esta separación entre contenido y forma permite que se pueda modificar el aspecto visual de un sitio Web sin afectar a los documentos ya creados y libera a los autores de preocuparse por el diseño final de sus páginas.

- **Presentación**

Un CMS puede gestionar automáticamente la accesibilidad de la Web, con soporte de normas internacionales de accesibilidad como WAI⁴, y adaptarse a las preferencias o necesidades de cada usuario. También puede proporcionar compatibilidad con los diferentes navegadores disponibles en todas las plataformas (Windows, Linux, Mac, Palm) y su capacidad de internacionalización lo permite adaptarse al idioma, sistema de medidas y cultura del visitante.

El sistema se encarga de gestionar muchos otros aspectos como son los menús de navegación o la jerarquía de la página actual dentro de la Web, añadiendo enlaces de forma automática. También gestiona todos los módulos, internos o externos, que incorpore al sistema. Así por ejemplo, con un módulo de noticias se presentarían las novedades aparecidas en otro sitio Web, con un módulo de publicidad se mostraría un anuncio o mensaje animado, y con un módulo de foro se podría mostrar, en la página principal, el título de los últimos mensajes recibidos. Todo eso con los enlaces correspondientes y, evidentemente, siguiendo el patrón que los diseñadores hayan creado.

⁴ Web Accessibility Initiative. Rama del World Wide Web Consortium que vela por la accesibilidad de la Web.

1.2.2 Necesidad de un CMS

En el apartado anterior se han presentado bastantes motivos para ver la utilidad de un sistema que gestione un entorno Web, pero se podría pensar que no es necesario para un sitio Web relativamente pequeño o cuando no se necesitan tantas funcionalidades. Eso sólo podría ser cierto para una Web con unas pocas páginas estáticas para el que no se prevea un crecimiento futuro ni muchas actualizaciones, lo que no es muy realista. En cualquier otro caso, la flexibilidad y escalabilidad que permiten estos sistemas, justifican su utilización en prácticamente cualquier tipo de Web.

Muchos usuarios particulares utilizan CMS gratuitos para elaborar y gestionar sus Webs personales, obteniendo Webs dinámicas llenas de funcionalidades. El resultado que obtienen es superior al de algunas empresas que se limitan a tener páginas estáticas que no aportan ningún valor añadido.

Éstos son algunos de los puntos más importantes que hacen útil y necesaria la utilización de un CMS:

- Inclusión de nuevas funcionalidades en la Web: Esta operación puede implicar la revisión de multitud de páginas y la generación del código que aporta las funcionalidades. Con un CMS eso puede ser tan simple como incluir un módulo realizado por terceros, sin que eso suponga muchos cambios en la Web. El sistema puede crecer y adaptarse a las necesidades futuras.
- Mantenimiento de gran cantidad de páginas: En una Web con muchas páginas hace falta un sistema para distribuir los trabajos de creación, edición y mantenimiento con permisos de acceso a las diferentes áreas. También se tienen que gestionar los metadatos de cada documento, las versiones, la publicación y caducidad de páginas y los enlaces rotos, entre otros aspectos.
- Reutilización de objetos o componentes: Un CMS permite la recuperación y reutilización de páginas, documentos, y en general de cualquier objeto publicado o almacenado.
- Páginas interactivas: Las páginas estáticas llegan al usuario exactamente como están almacenadas en el servidor Web. En cambio, las páginas dinámicas no existen en el servidor tal como se reciben en los navegadores, sino que se generan según las peticiones de los usuarios. De esta manera cuando por ejemplo se utiliza un buscador, el sistema genera una página con los resultados que no existían antes de la petición. Para conseguir esta interacción, los CMS conectan con una base de datos que hace de repositorio central de todos los datos de la Web.
- Cambios del aspecto de la Web: Si no hay una buena separación entre contenido y presentación, un cambio de diseño puede comportar la revisión de muchas páginas para su adaptación. Los

CMS facilitan los cambios con la utilización, por ejemplo, del estándar CSS⁵, con lo que se consigue la independencia de presentación y contenido.

- **Consistencia de la Web:** La consistencia en una Web no quiere decir que todas las páginas sean iguales, sino que hay un orden (visual) en vez de caos. Un usuario nota enseguida cuándo una página no es igual que el resto de las de la misma Web por su aspecto, la disposición de los objetos o por los cambios en la forma de navegar. Estas diferencias provocan sensación de desorden y dan a entender que el Web no lo han diseñado profesionales. Los CMS pueden aplicar un mismo estilo en todas las páginas con el mencionado CSS, y aplicar una misma estructura mediante patrones de páginas.
- **Control de acceso:** Controlar el acceso a una Web no consiste solo en permitir la entrada a la Web, sino que gestiona los diferentes permisos a cada área de la Web, aplicados a grupos o individuos.

1.2.3 Presente y futuro de los CMS

En la actualidad, además de la ampliación de las funcionalidades de los CMS, uno de los campos más interesantes es la incorporación de estándares que mejoran la compatibilidad de componentes, facilitan el aprendizaje al cambiar de sistema y aportan calidad y estabilidad. Algunos de estos estándares son: CSS, XML, XHTML⁶, WAI, RSS⁷.

También las aplicaciones que rodean los CMS acostumbran a ser estándar (de facto), como los servidores Web Apache e ISS; los lenguajes PHP, Perl y Python; y las bases de datos MySQL y PostgreSQL. La disponibilidad para los principales sistemas operativos de estas aplicaciones y módulos, permite que los CMS puedan funcionar en diversas plataformas sin muchas modificaciones.

“Los CMS se convertirán en un artículo de consumo, cuando los productos se hayan establecido y más soluciones lleguen al mercado. Eso provocará una disminución de los precios en los productos comerciales y una mayor consistencia en las funcionalidades que ofrecen.” (Robertson, 2003)

En este entorno:

- Muchas empresas que implementan Webs tendrán que cerrar.

⁵ Cascading Style Sheets. Lenguaje de programación que describe el estilo gráfico de un documento web.

⁶ Extensible Hypertext Markup Language. *Reformulación* del HTML 4.0 con XML 1.0. Toma la capacidad de formato del HTML y la une con la formalidad y extensibilidad del XML.

⁷ Really Simple Syndication. Formato XML formulado para canalizar las noticias de un sitio web especializado en noticias.

- Muchos proyectos fracasarán por no ajustarse a los estándares y no entender conceptos como usabilidad, arquitectura de la información, gestión del conocimiento y contenido.
- El campo de los gestores de contenido madurará hasta conseguir un alto grado de consistencia y profesionalismo.
- Se adoptarán estándares en el almacenaje, estructuración y gestión del contenido. Se producirá una fusión entre gestión de contenidos, gestión de documentos y gestión de registros.

1.2.4 CMS Drupal

Drupal es un CMS modular y muy configurable. Es un programa de código abierto, con licencia GNU/GPL⁸, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. Es un sistema multiusuario, multiplataforma, multilinguaje, extensible. Destaca por la calidad de su código y de las páginas generadas, el respeto de los estándares de la Web, y un énfasis especial en la usabilidad y consistencia de todo el sistema.

El diseño de Drupal es especialmente idóneo para construir y gestionar comunidades en Internet. Si bien las características básicas de Drupal son suficientes como para lanzar un sitio completo, el *core*⁹ incluye: administración de usuarios, páginas, vocabulario, comentarios y sindicación. Su potencialidad reside en su estructura modular que permite ser extendida y configurada a gusto y placer del administrador para distintas funciones por medio de módulos de fácil instalación.

Introduce el concepto de nodo como sinónimo de tipos de contenido, cualquier recurso que se ingrese al sistema pasa a ser un nodo, que puede ser variable e incluir artículos, historias, *posts*, encuestas, imágenes, libros colaborativos, reseñas, recetas, entre otros. Este nuevo concepto nos permite estandarizar la información asignándoles las mismas características a distintos objetos y la posibilidad de tener toda la información catalogada. La clasificación de los nodos es un tanto arbitraria y está al servicio de los usuarios/administradores de contenidos que pueden construir los que necesite.

El sistema de taxonomía de Drupal es robusto y potente, el contenido se organiza en categorías que se arman a través del módulo taxonomías, el cual permite generar vocabularios controlados con términos que pueden ordenarse jerárquicamente y asociarse a un tipo de nodo en particular si hiciera falta. Se puede

⁸ GNU General Public License. Licencia de software libre ampliamente usada.

⁹ Núcleo.

configurar que un nodo pueda ser clasificado bajo uno o múltiples términos de un vocabulario, flexibilidad que agradecen quienes administran sistemas de contenidos rígidos.

Algunas de sus características generales son:

- Ayuda on-line: Un robusto sistema de ayuda online y páginas de ayuda para los módulos del núcleo, tanto para usuarios como para administradores.
- Búsqueda: Todo el contenido en Drupal es totalmente indexado en tiempo real y se puede consultar en cualquier momento.
- Código abierto: El código fuente de Drupal está libremente disponible bajo los términos de la licencia GNU/GPL. Al contrario que otros sistemas de *blogs* o de gestión de contenido propietarios, es posible extender o adaptar Drupal según las necesidades.
- Módulos: La comunidad de Drupal ha contribuido muchos módulos que proporcionan funcionalidades como página de categorías, autenticación mediante *jabber*, mensajes privados, *bookmarks*.
- Personalización: Un robusto entorno de personalización está implementado en el núcleo de Drupal. Tanto el contenido como la presentación pueden ser individualizados de acuerdo las preferencias definidas por el usuario.
- Autenticación de usuarios: Los usuarios se pueden registrar e iniciar sesión de forma local o utilizando un sistema de autenticación externo u otro sitio Drupal. Para su uso en una Intranet, Drupal se puede integrar con un servidor LDAP¹⁰.
- Permisos basados en roles: Los administradores de Drupal no tienen que establecer permisos para cada usuario. En lugar de eso, pueden asignar permisos a un rol y agrupar los usuarios por roles.
- Plantillas (Templates): El sistema de temas de Drupal separa el contenido de la presentación permitiendo controlar o cambiar fácilmente el aspecto del sitio Web. Se pueden crear plantillas con HTML y/o con PHP.
- Agregador de noticias: Drupal incluye un potente Agregador de Noticias para leer y publicar enlaces a noticias de otros sitios Web. Incorpora un sistema de cache en la base de datos, con temporización configurable.

¹⁰ Lightweight Directory Access Protocol. Protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

- Independencia de la base de datos: Aunque la mayor parte de las instalaciones de Drupal utilizan MySQL, existen otras opciones. Drupal incorpora una capa de abstracción de base de datos que actualmente está implementada y mantenida para MySQL y PostgreSQL, aunque permite incorporar fácilmente soporte para otras bases de datos.
- *Multiplataforma*: Drupal ha sido diseñado desde el principio para ser multiplataforma. Puede funcionar con Apache o Microsoft IIS como servidor Web y en sistemas como Linux, BSD, Solaris, Windows y Mac OS X. Por otro lado, al estar implementado en PHP, es totalmente portable.
- Múltiples idiomas y Localización: Drupal está pensado para una audiencia internacional y proporciona opciones para crear un Portal multilingüe. Todo el texto puede ser fácilmente traducido utilizando una interfaz Web, importando traducciones existentes o integrando otras herramientas de traducción.
- Administración vía Web: La administración y configuración del sistema se puede realizar enteramente con un navegador y no precisa de ningún software adicional.
- Foros de discusión: Drupal incorpora foros de discusión para crear sitios comunitarios vivos y dinámicos.
- Control de congestión: Drupal incorpora un mecanismo de control de congestión que permite habilitar y deshabilitar determinados módulos o bloques dependiendo de la carga del servidor. Este mecanismo es totalmente configurable y ajustable.
- Sistema de Cache: El mecanismo de cache elimina consultas a la base de datos incrementando el rendimiento y reduciendo la carga del servidor.

1.3 Metodologías de Desarrollo

“Durante los últimos años han surgido diferentes metodologías que se mueven en entornos concretos dando más importancia a los aspectos más importantes de los sistemas que tratan” (Escalona, 2001).

Para facilitar el trabajo de los desarrolladores de software se han propuesto metodologías que ofrecen modelos y técnicas apropiadas para realizar software, y algunas para desarrollar aplicaciones web en específico, éstas metodologías han tenido gran aceptación dentro de la comunidad global. A continuación se especifican algunas de estas metodologías.

1.3.1 Web Site Design Method (WSDM)

WSDM es un método de diseño para aplicaciones Web que define el sistema en base a los grupos de usuarios. Consta de cinco fases de diseño, teniendo como base de partida un compromiso explícito de los usuarios y termina con la aplicación real.

Más que otros métodos, WSDM es una metodología, pues no sólo proporciona primitivas de modelado que permiten un desarrollador Web para construir modelos que describen el sitio Web o la aplicación de diferentes perspectivas y en diferentes niveles de abstracción, sino que también proporciona una manera sistemática para desarrollar la aplicación Web.

En la figura 1, se pueden ver las cinco fases de diseño, y los modelos de salida que se produce en cada fase.

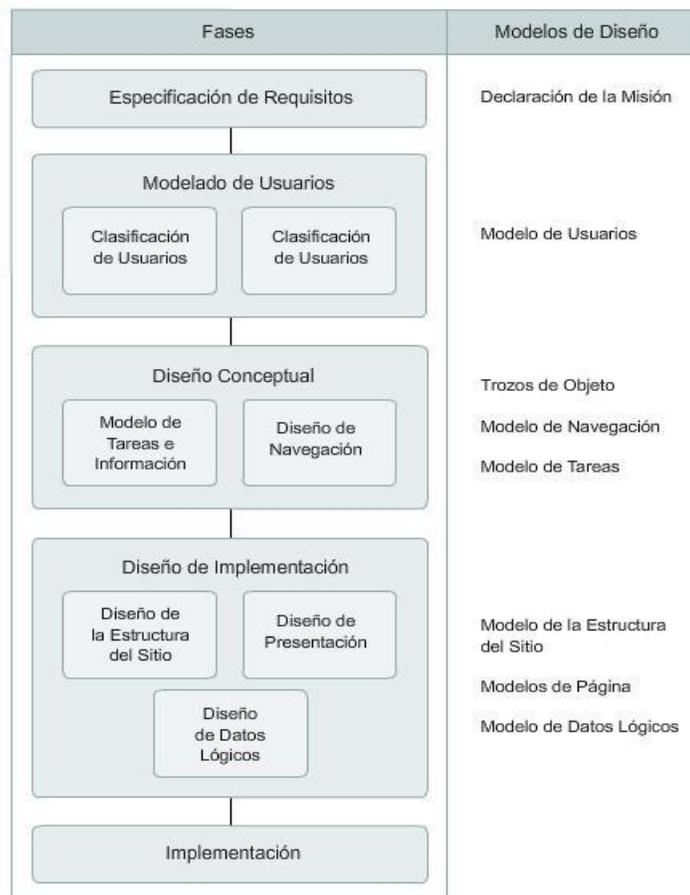


Figura 1. Esquema de fases de WSDM

Fase 1: Especificación de Requisitos

En esta primera fase se especifican el propósito, el objetivo y los usuarios establecidos para la página Web. Establece las fronteras del proceso de diseño. Además, la Especificación de Requisitos es útil para la validación, es decir, para comprobar si el sitio Web ha alcanzado el objetivo formulado.

Fase 2: Modelado de Usuarios

Los usuarios identificados en la Especificación de Requisitos son refinados mediante dos sub-etapas: Clasificación de los Usuarios y Caracterización de los Usuarios. Durante la Clasificación de los Usuarios se identifican los distintos tipos de usuarios con mayor detalle y se clasifican agrupándolos según sus requisitos de información y funcionalidades en grupos de usuarios. Durante la Caracterización de los Usuarios se especifican las características de cada grupo de usuarios.

Fase 3: Diseño Conceptual

El objetivo de esta fase es convertir los requisitos informales obtenidos en fases anteriores en descripciones formales con alto grado de precisión que pueden ser utilizados más adelante para generar el sistema Web. Durante el Modelo de Tareas e Información se modela en detalles las tareas que los miembros de cada grupo de usuarios tienen que ser capaces de realizar y formalmente se describen los datos y la funcionalidad que se necesitan para esas tareas. El objetivo del Diseño de Navegación es definir la estructura conceptual del sistema Web y modelar como los miembros de los diferentes grupos de usuarios pueden navegar por el sistema y realizar sus tareas.

Fase 4: Diseño de Implementación

Se complementa el diseño conceptual con los detalles necesarios para la implementación mediante 3 subfases. Durante el Diseño de la Estructura del Sitio el diseñador decide cómo los componentes del Modelo de Navegación serán agrupados en páginas. En el Diseño de Presentación se definen el aspecto de la Web y el diseño de las páginas teniendo en cuenta su posicionamiento y sus elementos. En el Diseño Datos Lógicos se crea el esquema lógico de datos que será utilizado por el sistema Web.

Fase 5: Implementación

La implementación puede ser generada automáticamente a partir de toda la información recogida en fases anteriores.

1.3.2 UML-Based Web Engineering (UWE)

Es una metodología de desarrollo de aplicaciones Web, prestando especial atención en sistematización y personalización.

UWE es una propuesta basada en *Rational Unified Process* (RUP) y UML¹¹ pero adaptados a la Web. En requisitos separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito. Considera los requisitos de navegación como un tipo de requisito funcional y, aunque no propone técnicas específicas para el tratamiento de los mismos, principalmente se basa en los casos de uso, los que separa con idea de identificar mejor los aspectos que influirán en el Modelo de Navegación.

En la figura 2 se muestran las tres fases del procedimiento de desarrollo de UWE:



Figura 2: Esquema de fases de UWE

Consiste en una notación y en un método. La notación se basa en UML para aplicaciones Web en general y para aplicaciones adaptativas en particular.

¹¹ Unified Modeling Language. Notación gráfica utilizada para describir sistemas de software.

Fase 1: Captura de Requisitos

Se identifican los usuarios y se obtienen los requisitos; se describen los usuarios del sistema, las reglas de adaptación los casos de uso y las interfaces de la aplicación Web, teniendo como resultado el Modelo de Casos de Uso.

Fase 2: Análisis y Diseño

Se realiza un diseño detallado de las clases y se definen subsistemas e interfaces, obteniendo el Modelo Conceptual, el Modelo de Usuario, el Modelo de Navegación, el Modelo de Estructura de Presentación, Modelo de Adaptación y el Modelo de la Arquitectura.

Fase 3: Implementación

Se implementa la arquitectura, el modelo de usuario, las interfaces de usuario, los mecanismos adaptativos y las tareas referentes a la integración de todas estas implementaciones.

1.3.3 Navigational Development Techniques (NDT)

El proceso de NDT se centra en una detallada fase de ingeniería de requisitos guiada por objetivos, que contempla tanto la captura, como la definición y la verificación de requisitos.

El proceso comienza definiendo los objetivos y en base a éstos se describe un proceso por el que se pueden capturar y definir los diferentes requisitos del sistema. Éstos son clasificados y tratados dependiendo de la tipología a la que pertenezcan.

NDT divide los requisitos en:

- Requisitos de almacenamiento de información, que definen qué información se va a manejar en el sistema y cómo se relacionan entre sí.
- Requisitos de actores, en los que se definen los roles que podrán interactuar en el sistema y las relaciones que se puede producir entre ellos.
- Requisitos funcionales, que permitirán definir la funcionalidad del sistema.
- Requisitos de interacción, que definen cómo se muestra la información, cómo se podrá navegar en el sistema y los criterios de recuperación que se ofrecen.
- Requisitos no funcionales, que recogerán otros requisitos del sistema.

Una vez validados estos requisitos, el proceso de NDT propone generar tres modelos: el Modelo Conceptual, que representa mediante un diagrama de clases la estructura estática del sistema; el Modelo de Navegación, que representa mediante un conjunto de diagramas de clases la forma en que se podrá navegar en el sistema; y el Modelo de Interfaz Abstracta, que mediante un conjunto de prototipos evaluables permitirá mostrar cómo se va a interactuar con el sistema.

La característica más destacable del proceso propuesto por NDT es que el paso de especificación de requisitos a estos modelos se hace de una manera sistemática e independiente.

El proceso de NDT está totalmente basado en la definición de requisitos. Si se entra más en detalle, se verá que a su vez, el proceso de definición de requisitos se basa mucho en los diferentes roles de usuario que puedan aparecer en el sistema.

NDT no contempla las fases avanzadas del ciclo de vida del proyecto, como podrían ser implementación o mantenimiento. En NDT estos flujos de trabajo se realizan de una forma sistemática, como la generación de los modelos básicos, se pueden realizar de manera automática si se hace uso de una herramienta CASE que acompaña a NDT: NDT-Tool.

1.3.4 Web Modeling Language (WebML)

WebML es una notación visual para el diseño de aplicaciones Web complejas que usan datos intensivamente. Provee especificaciones gráficas formales para un proceso de diseño completo que puede ser asistido por herramientas de diseño visuales.

Los principales objetivos del proceso de diseño WebML son:

- Expresar la estructura de una aplicación Web con un alto nivel de descripción.
- Proveer múltiples vistas de un mismo contenido.
- Almacenar toda la información recogida durante el proceso de diseño en un repositorio, la cual puede ser utilizada por páginas Web generadas dinámicamente.
- Modelar usuarios y comunidades explícitamente en el repositorio.
- Permitir la especificación de las operaciones de manipulación de la información para actualizar el contenido del sitio o interactuar con servicios externos.

En la siguiente figura se muestran las fases de desarrollo de WebML:

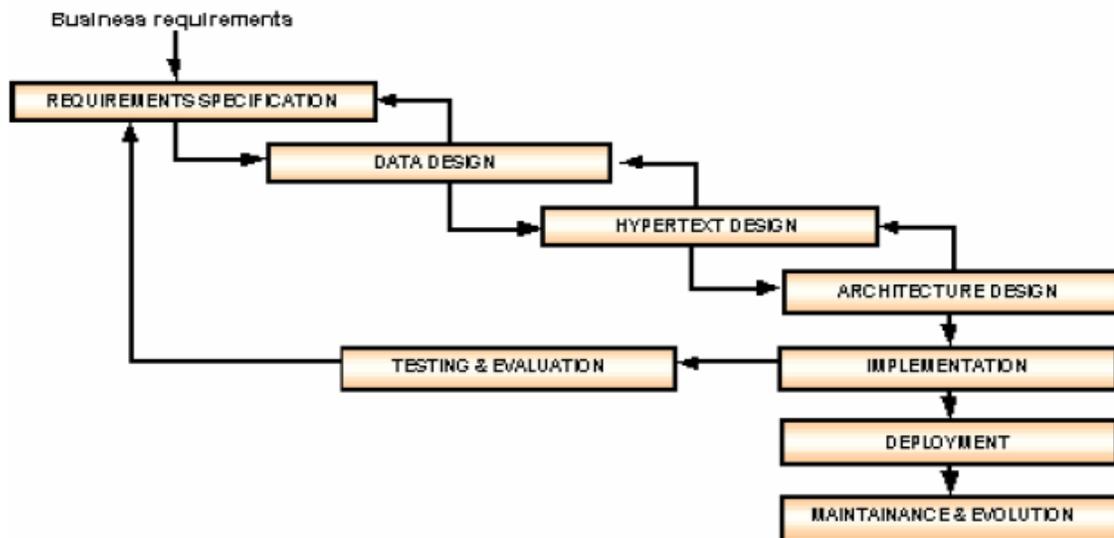


Figura 3: Esquema de fases de WebML

WebML propone los siguientes modelos:

- Modelo Conceptual: Representa la estructura estática del sistema.
- Modelo de Hipertexto¹²: Describe los hipertextos que pueden ser publicados en el sitio Web. La descripción de los hipertextos se realiza mediante los Modelos de Composición, que define las páginas que componen el sistema y el Modelo de Navegación, que describe como se podrá navegar a través de ellas.
- Modelo de Presentación: Define la apariencia física de las páginas.
- Modelo de Personalización: Se identifican los diferentes usuarios y sus necesidades de personalización. Se capturan los requisitos funcionales y no funcionales.

WebML se enfoca en el diseño de la interfaz. Para esto provee una serie de estereotipos que pueden ser implementados usando XML. Permite a los modeladores expresar de forma conceptual los datos y la navegación de una aplicación Web utilizando diagramas Entidad-Relación¹³.

¹² Vínculo en una o varias palabras de un texto o una imagen que remite a otra página web donde se puede encontrar información complementaria.

¹³ Herramienta para el modelado de datos de un sistema de información

1.3.5 Rational Unified Process (RUP)

RUP es un proceso de desarrollo de software y unido a UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Está enfocado hacia el cliente.

RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización.

La metodología RUP es más apropiada para proyectos grandes, dado que requiere un equipo de trabajo capaz de administrar un proceso complejo en varias etapas.

RUP describe como diseñar una arquitectura flexible, que se acomode a los cambios, comprensible intuitivamente y promueve una más efectiva reutilización de software. Soporta el desarrollo de software basado en componentes: módulos no triviales que completan una función clara. RUP provee un enfoque sistemático para definir una arquitectura utilizando componentes nuevos y preexistentes.

RUP muestra como modelar software visualmente para capturar la estructura y comportamiento de arquitecturas y componentes. Las abstracciones visuales ayudan a comunicar diferentes aspectos del software; comprender los requerimientos, ver como los elementos del sistema se relacionan entre sí, mantener la consistencia entre diseño e implementación y promover una comunicación precisa.

El aseguramiento de la calidad se construye dentro del proceso, en todas las actividades, involucrando a todos los participantes, utilizando medidas y criterios objetivos, permitiendo así detectar e identificar los defectos en forma temprana.

Es también una guía para establecer espacios de trabajo seguros para cada desarrollador, suministrando el aislamiento de los cambios hechos en otros espacios de trabajo y controlando los cambios de todos los elementos de software.

RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades. En la siguiente figura se muestran las fases y los flujos de trabajos de RUP. Se muestra cómo varía el esfuerzo asociado a las disciplinas según la fase en la que se encuentre el proyecto RUP.

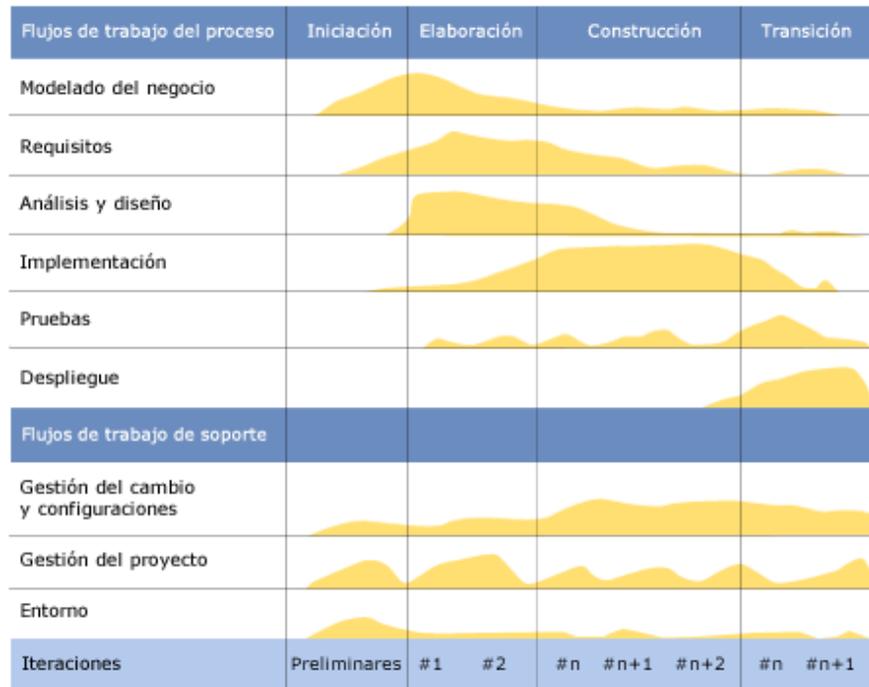


Figura 4: Esquema de fases e iteraciones de RUP

Las primeras iteraciones en las fases de *Inicio* y *Elaboración* se enfocan hacia la comprensión del problema y la tecnología, la delimitación del ámbito del proyecto, la eliminación de los riesgos críticos, y al establecimiento de una línea base de la arquitectura.

Durante la fase de *Inicio* las iteraciones hacen mayor énfasis en actividades de modelado del negocio y de requerimientos.

En la fase de *Elaboración*, las iteraciones se orientan al desarrollo de la línea base de la arquitectura, abarcan más los flujos de trabajo de requerimientos, modelo de negocios (refinamiento), análisis, diseño y una parte de implementación orientado a la línea base de la arquitectura.

En la fase de *Construcción*, se lleva a cabo la construcción del producto por medio de una serie de iteraciones.

Para cada iteración se selecciona algunos casos de uso, se refina su análisis y diseño y se procede a su implementación y pruebas. Se realizan tantas iteraciones hasta que se termine la implementación de la nueva versión del producto.

En la fase de *Transición* se pretende garantizar que se tiene un producto preparado para su entrega a la comunidad de usuarios.

Como se puede observar en cada fase participan todas las disciplinas, pero que dependiendo de la fase el esfuerzo dedicado a una disciplina varía.

1.3.6 eXtreme Programming (XP)

XP es la más popular dentro de las metodologías ágiles. XP está centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. Se basa en una serie de valores que se aplican una y otra vez en las distintas prácticas utilizadas en la programación extrema: realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

“Muchas de las prácticas que propone XP son técnicas antiguas, tratadas y probadas, aunque a menudo olvidadas por muchos, incluyendo la mayoría de los procesos planeados. Además de resucitar estas técnicas, la XP las teje en un todo sinérgico donde cada una refuerza a las demás” (Fowler, 2000).

Fases y prácticas XP

- Planificación del proyecto

Historias de usuario: El primer paso de cualquier proyecto que siga la metodología XP es definir las historias de usuario con el cliente. Las historias de usuario tienen la misma finalidad que los casos de uso pero con algunas diferencias: constan de tres o cuatro líneas escritas por el cliente en un lenguaje no técnico sin hacer mucho hincapié en los detalles; no se debe hablar ni de posibles algoritmos para su implementación ni de diseños de base de datos adecuados.

Son usadas para estimar tiempos de desarrollo de la parte de la aplicación que describen. También se utilizan en la fase de pruebas, para verificar si el programa cumple con lo que especifica la historia de usuario. Cuando llega la hora de implementar una historia de usuario, el cliente y los desarrolladores se reúnen para concretar y detallar lo que tiene que hacer dicha historia. El tiempo de desarrollo ideal para una historia de usuario es entre una y tres semanas.

Release planning: Después de tener ya definidas las historias de usuario es necesario crear un plan de publicaciones donde se indican las historias de usuario que se crearán para cada versión del programa y las fechas en las que se publicarán estas versiones. Un *release plan* es una planificación donde los desarrolladores y clientes establecen los tiempos de implementación ideales de las historias de usuario, la prioridad con la que serán implementadas y las historias que serán implementadas en cada versión del programa.

Iteraciones: Todo proyecto que siga la metodología XP se ha de dividir en iteraciones de aproximadamente tres semanas de duración. Al comienzo de cada iteración los clientes deben seleccionar las historias de usuario definidas en el *release planning* que serán implementadas. También se seleccionan las historias de usuario que no pasaron el test de aceptación que se realizó al terminar la iteración anterior. Estas historias de usuario son divididas en tareas de entre uno y tres días de duración que se asignarán a los programadores.

Velocidad del proyecto: Medida que representa la rapidez con la que se desarrolla el proyecto. Para estimarla hay que contar el número de historias de usuario que se pueden implementar en una iteración, de esta forma se sabrá la cantidad de historias que se pueden desarrollar en las distintas iteraciones. Usando la velocidad del proyecto se controla que todas las tareas se puedan desarrollar en el tiempo del que dispone la iteración.

Reuniones diarias: Es necesario que los desarrolladores se reúnan diariamente y expongan sus problemas, soluciones e ideas de forma conjunta. Las reuniones tienen que ser fluidas y todo el mundo tiene que tener voz y voto.

- Diseño

Diseños simples: Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto.

Metáfora: El sistema es definido mediante una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo. Una metáfora es una historia compartida que describe cómo debería funcionar el sistema.

Funcionalidad extra: Nunca se debe añadir funcionalidad extra al programa aunque se piense que en un futuro será utilizada.

Spike solutions: Programa muy simple para explorar soluciones potenciales y para arreglar las respuestas a los problemas técnicos o del diseño que sean resistentes.

Refactorizar: Es una técnica para reestructurar el código, alterando su estructura interna sin alterar el comportamiento externo con el objetivo de remover duplicación de código. “Su corazón es una serie de pequeños comportamientos que preservan las transformaciones, cada transformación (llamada refactorización) hace poco, pero una secuencia de transformaciones puede producir una reestructuración significativa” (Fowler, 2000).

Tarjetas C.R.C.: El uso de las tarjetas C.R.C (*Class, Responsibilities and Collaboration*) permiten al programador centrarse y apreciar el desarrollo orientado a objetos olvidándose de los malos hábitos de la programación procedimental clásica. Las tarjetas C.R.C representan objetos; la clase a la que pertenece el objeto se puede escribir en la parte de arriba de la tarjeta, en una columna a la izquierda se pueden escribir las responsabilidades u objetivos que debe cumplir el objeto y a la derecha, las clases que colaboran con cada responsabilidad.

- Codificación

Disponibilidad del cliente: El cliente real(es el que usará el sistema) tiene que estar con el equipo de trabajo, debe responder preguntas, resolver disputas, establecer prioridades, discutir mejoras. La comunicación oral es más efectiva que la escrita.

Estándares de programación: XP plantea que es indispensable que se sigan ciertos estándares de programación para mantener un mismo estilo de código: homogéneo y legible. Son fundamentales cuando los programadores cambian de pareja o hacen refactorización del código de otros.

Programación en parejas: Todo el código se escribe en parejas, se produce código de mayor calidad, se extiende el conocimiento y se realiza el trabajo de una persona en casi la mitad del tiempo y mejor.

Pruebas: La producción de código está dirigida por las pruebas unitarias. Estas son establecidas por el cliente antes de escribirse el código y son ejecutadas constantemente ante cada modificación del sistema.

Integración continua: Cada pieza de código es integrada en el sistema una vez que esté lista y se prueba. Así, el sistema puede llegar a ser integrado y construido varias veces en un mismo día.

Propiedad colectiva del código: Cualquiera puede modificar el código en cualquier momento. Se evitan cuellos de botella en la codificación, todos asumen las responsabilidades sobre el conjunto del sistema.

- Prueba

Pruebas: La producción de código está dirigida por las pruebas unitarias. Estas son establecidas por el cliente antes de escribirse el código y son ejecutadas constantemente ante cada modificación del sistema.

Pruebas de aceptación: Son un sistema de prueba de caja negra. Cada prueba de aceptación representa algún resultado esperado del sistema. Los clientes son los responsables de verificar que las pruebas estén correctas para decidir cuál prueba fallida tiene la prioridad más alta.

1.4 Análisis de las Metodologías Anteriores

Todas estas metodologías contienen las fases clásicas de un proceso de desarrollo: especificación de requisitos, análisis y diseño, implementación y prueba; algunas profundizan el trabajo en una fase o varias fases y le prestan menos importancia a otras, pero de manera general todas son tomadas en cuenta en todos los métodos analizados.

La fase de especificación de requisitos es tratada en todas las metodologías analizadas, aunque cada una de ellas utiliza métodos distintos. El proceso de NDT se centra en una detallada fase de ingeniería de requisitos guiada por objetivos mientras que la metodología WSDM, utiliza el lenguaje natural para la definición de requisitos. La técnica de casos de usos la utilizan RUP, WebML y UWE; este último también utiliza las técnicas de escenarios y glosarios. eXtreme Programming (XP) define historias de usuarios con el cliente, estas tienen la misma finalidad que los casos de usos pero con algunas diferencias. Viendo esta fase en cada una de las metodologías se puede llegar a la conclusión que la técnica de los casos de usos es la más aceptada en la fase de especificación de requisitos.

La fase de análisis es tratada en casi todos los procesos analizados. WSDM llama esta fase *Modelado de Usuario*, donde define dos subfases que centran su esfuerzo en el análisis del estudio de los grupos de usuarios. RUP y UWE unen las fases de análisis y diseño en una sola, en el caso de RUP el resultado final del flujo de trabajo del análisis es el modelo del análisis que es un modelo de objetos conceptual, el modelo incluye los paquetes, clases y realizaciones de casos de uso del análisis, y la vista de la arquitectura del modelo de análisis; mientras que UWE define entre otros el modelo conceptual, de usuario, de navegación, de presentación, de adaptación y de la arquitectura. WebML contempla en esta fase el modelo conceptual y el de hipertexto. Analizando esta fase en cada uno de los procesos se puede inferir que todos estos métodos coinciden en la realización de un modelo conceptual dentro de la fase de análisis, incluyendo la metodología NDT, que sin presentar una fase de análisis propone la realización de este modelo.

La fase de diseño es tratada en todos los métodos analizados, aunque de forma diferente. En el método WSDM se hace énfasis en la importancia de la modelación de la información y los requerimientos funcionales, así como la modelación estructural del sitio Web; se complementa el diseño conceptual con los detalles necesarios para la implementación. En RUP el diseño describe cómo el sistema será realizado a partir de los requerimientos, indica con precisión lo que se debe programar, se basa en los artefactos y las actividades para dar cumplimiento a los requisitos obtenidos. El proceso NDT propone junto al modelo conceptual, el modelo de navegación y el modelo de interfaz abstracta. WebML se centra en el diseño de interfaz y propone los modelos de presentación. XP propone realizar soluciones simples, define el sistema en una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo, y refactorizar, esta práctica es importante porque su objetivo es eliminar la duplicación de código, esto se logra alterando la estructura interna del mismo sin alterar el comportamiento externo.

La fase de implementación es tratada en las metodologías WSDM, NDT, RUP y XP. En los métodos WSDM y NDT y WebML no le asignan importancia a esta fase, sugieren generar el código automáticamente, en el caso de la metodología NDT mediante la herramienta CASE: NDT Tools; el método WebML dispone de la herramienta WebRatio que facilita la implementación de páginas Web en jsp¹⁴. En UWE se implementa la arquitectura, el modelo de usuario, las interfaces de usuario y los mecanismos adaptativos. En la fase de implementación de la metodología RUP se define cómo se organizan las clases y objetos en componentes, cuáles nodos se utilizarán y la ubicación en ellos de los componentes, y la estructura de capas de la aplicación. En XP se promueve la simplicidad de las soluciones implementadas y aplica varias prácticas, como son:

- Disponibilidad del cliente: expresa que el cliente debe formar parte del equipo de trabajo, debe responder preguntas, establecer prioridades.
- Programación en parejas: se produce un código de mayor calidad y se realiza el trabajo en casi la mitad del tiempo y mejor.
- Pruebas: deben ser ejecutadas constantemente ante cada modificación del sistema, logrando una corrección de errores en tiempo real.
- Integración continua: cada pieza de código debe ser integrada en el sistema cada vez que esté lista y se prueba, así el sistema puede ser integrado y construido varias veces en un mismo día.

¹⁴ Java Server Pages. Tecnología que permite la creación de documentos HTML de forma dinámica en un servidor.

La manera en que XP trata la fase de implementación es la más idónea para la realización del proceso de desarrollo.

Después de haber analizado todas estas metodologías se puede concluir que:

- Las metodologías WSDM, UWE, WebML y NDT pueden servir de apoyo pues plantean conceptos interesantes y eficientes, aunque son demasiado extensas y no tienen en cuenta la fase de prueba, la cual es muy importante para asegurar la calidad del producto final. Varias de estas metodologías no describen el proceso de implementación del sistema de manera convincente, pues algunas proponen que la implementación se haga de forma automática. La documentación de estos métodos generalmente se basa en criterios y valoraciones de otros autores.
- RUP es una metodología que va creando durante todo el proceso de desarrollo una sólida documentación sobre todo el trabajo que se va realizando. Debido a que propone que se realicen gran cantidad de actividades en cada flujo de trabajo y que tiene muchos artefactos y trabajadores, esta metodología es más apropiada para proyectos grandes.
- XP es una metodología ágil, es el método más adecuado para utilizar en grupos pequeños de trabajo, en los que se mantiene una constante comunicación con los clientes y está especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, pero esta metodología no lleva una documentación consistente durante su proceso de desarrollo y en caso de ocurrir cualquier resistencia del cliente o del equipo de desarrollo hacia las prácticas y principios que propone XP, el proceso puede fracasar llevando consigo altas comisiones; estas desventajas hacen que la metodología no sea del todo eficiente.

Después de haber obtenido los elementos de estas metodologías que pueden ser útiles para el proceso de desarrollo, así como haber analizado el porqué estos métodos no son totalmente eficientes, es que se decide definir un procedimiento de trabajo para el desarrollo de Portales Web utilizando Drupal que tenga las siguientes características:

- Ciclo de trabajo completo: El proceso de desarrollo está organizado en 4 fases que cubren todo el ciclo de vida del proyecto. Abarca todas las etapas de construcción del software, comenzando con la modelación del negocio y la captura de requisitos y terminando con la implantación y publicación del producto final.

- Equipo de desarrollo dividido en roles: Los integrantes del equipo de trabajo encargado de realizar la aplicación Web, están organizados de tal manera, que cada uno tiene un rol definido, con sus propias características y tareas a realizar, para el cual recibe una capacitación previa.
- Tareas establecidas para cada fase del ciclo de trabajo: Dentro de cada fase se definen tareas que complementan la efectiva culminación de la fase, los integrantes del equipo de trabajo dependiendo de su rol, tienen bien especificados cuales son los objetivos que deben cumplir.
- Documentación en cada fase: Al culminar cada fase, son producidos, modificados o utilizados documentos, modelos, código de fuente, archivos, entre otros. El proceso de desarrollo debe orientar en cada fase cuales de estos elementos se deben obtener.

Al definir el procedimiento de trabajo, se pretende realizar Portales Web, con un diseño sencillo y claridad en el código, que cumplan con las expectativas del cliente y que sean terminados dentro del tiempo establecido en el contrato.

1.5 Conclusiones

Tras haber analizado todo lo que se muestra en este capítulo, se puede llegar a la conclusión de que el uso de un CMS, y en particular de Drupal, facilita el trabajo para realizar aplicaciones Web, debido a sus características y ventajas. Debido a la complejidad del trabajo con Drupal es necesario utilizar una metodología que guíe el trabajo con esta herramienta, por lo que fueron analizadas metodologías de desarrollo Web que se aplican hoy en día, viendo sus ventajas y desventajas, y tomando varios aspectos que contribuirán a la construcción de procedimiento de desarrollo, objetivo de este trabajo.

CAPÍTULO 2: PROCEDIMIENTO DE TRABAJO

2.1 Introducción

En este capítulo se muestran todos los roles involucrados en el procedimiento de trabajo, así como las tareas que deben cumplir, de manera general, durante la construcción del Portal Web. Se definen cuatro fases de desarrollo que abarcan todo el proceso de creación de una aplicación Web, desde la planificación hasta la implantación y publicación del Portal Web. Se establecen las actividades a desarrollar en cada fase y se especifica que trabajador debe realizar cada una de estas actividades. Además se definen los artefactos que se deben obtener al concluir cada fase de desarrollo; las plantillas de estos artefactos se encuentran en el Expediente de Calidad de Proyectos de la Universidad. En los anexos del trabajo se incluyen las plantillas de los artefactos que no se encuentran en el expediente y que son definidos en el procedimiento de trabajo.

2.2 Roles involucrados en el procedimiento de trabajo

En el desarrollo de un Portal Web están involucrados un grupo de roles necesarios para favorecer la realización de un producto con la calidad requerida. Estos roles son:

- Líder de la solución de software: es el responsable de todo el desarrollo del Portal Web, planifica y guía las actividades que se lleven a cabo.
- Analista: es el encargado de identificar y conducir los requisitos del Portal Web a lo largo del desarrollo.
- Arquitecto de Información: es el encargado de organizar, estructurar y distribuir la información de las páginas del Portal Web de manera que facilite la usabilidad del mismo.
- Diseñador gráfico: es el responsable de realizar el diseño visual del Portal Web atendiendo a aspectos tipográficos y cromáticos.
- Desarrolladores: son los encargados de implementar el Portal Web. Entre los desarrolladores están: programadores y diseñadores de base de datos:
 - Programador: es el encargado de convertir la especificación del sistema en código ejecutable.
 - Diseñador de base de datos: es el encargado de diseñar la base de datos que será usada por el Portal Web.

- Equipo de calidad: es el encargado de realizar las pruebas de calidad al producto, de manera que se asegure la calidad desde los inicios del proyecto.
- Administrador del Portal Web: es el encargado de configurar y administrar el CMS de forma que responda a las pautas del diseño y las funcionalidades con que debe contar el Portal Web.
- Psicólogo organizacional: es el encargado del estudio de los fenómenos psicológicos en el interior de las organizaciones y de cómo los procesos organizacionales fluyen.
- Equipo de despliegue: es el encargado de instalar y configurar el Portal Web en un ambiente real.
- Capacitador: es el encargado de impartir cursos de capacitación al usuario final, específicamente a los usuarios que administrarán el Portal Web.

2.3 Fases del procedimiento de trabajo

La propuesta de solución tiene cuatro fases de desarrollo: Inicio (RUP, 2003); Definiciones para el desarrollo; Diseño, implementación y pruebas e Implantación y publicación. Cada fase concluye con un hito fundamental. La siguiente figura muestra las fases y los hitos de cada una de ellas.

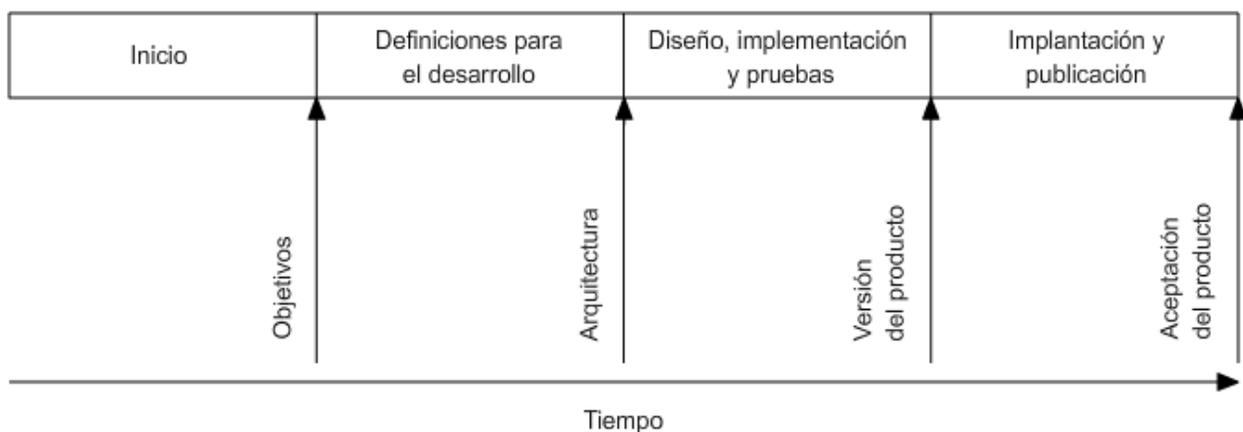


Figura 5: Fases e hitos del desarrollo de Portales Web basados en el CMS Drupal.

A continuación se definen los objetivos generales de cada fase, los trabajadores involucrados, así como las actividades que se realizan y los artefactos que se generan en cada fase de desarrollo del

procedimiento de trabajo. Debido a que los términos: trabajador, artefacto y actividades son ampliamente utilizados se incluyen además las definiciones de dichos términos.

Trabajador: representa cada uno de los roles involucrados en la creación de la aplicación Web; “es un papel que un individuo puede desempeñar en el desarrollo de un software (...)” (Jacobson y otros, 2000). Puede representar un grupo de personas que trabajan juntas, por ejemplo el Equipo de Calidad. Una persona puede ser varios trabajadores durante el ciclo de vida de un proyecto.

Artefacto: “es un producto tangible resultante del proceso de desarrollo del software” (RUP, 2003); se refiere a la documentación generada a lo largo del desarrollo del producto, aunque en ocasiones un artefacto puede referirse a un producto terminado. Algunos artefactos ayudan a la descripción de la función, la arquitectura o el diseño del software, mientras que otros se enfocan en el proceso de desarrollo en sí mismo. Los artefactos que detallan el diseño pretendido suelen realizarse al principio del proyecto y no necesitan mantenerse, mientras que otros se realizan durante el ciclo de vida con información que se actualiza durante el desarrollo.

Actividades: son todas las tareas, que manteniendo un orden lógico se realizan para darle cumplimiento a los objetivos de cada fase del procedimiento de trabajo.

2.3.1 Fase 1: Inicio

La fase de inicio es esencial en la planificación del desarrollo del Portal Web definiendo los entregables, cronogramas, personas involucradas y recursos necesarios para el ambiente de trabajo y la infraestructura tecnológica para un entorno real.

Objetivos de la fase de Inicio:

- Establecimiento de las etapas de desarrollo, costos y cronogramas del Portal Web.
- Estimación de riesgos.
- Definición del ambiente de desarrollo, recursos humanos e infraestructura tecnológica necesaria para la puesta en marcha del Portal Web.
- Capacitación del equipo de desarrollo en el lenguaje de programación, gestor de Base de Datos, plataforma y herramientas necesarias para el desarrollo del Portal Web.

Trabajadores involucrados en la fase de Inicio:

- Líder de la solución del software: es el encargado de concebir el proyecto, y preparar el ambiente de trabajo y recursos humanos. Estable los costos, etapas de desarrollo y cronogramas, y organiza la capacitación del equipo de desarrollo. Define la lista de riesgos del proyecto, así como el alcance y los objetivos del Portal Web.
- Arquitecto de información: es el encargado de definir la audiencia del Portal Web.

Actividades de la fase de Inicio:

- Concebir el proyecto: el equipo de desarrollo se reúne con el cliente para llegar a un acuerdo sobre cuáles son las tareas que tiene que desarrollar cada parte. El Líder de la solución de software realiza el Contrato, artefacto que deja establecido que el equipo de desarrollo y el cliente deben trabajar de forma conjunta para lograr el desarrollo del Portal Web. El Líder de la solución de software elabora el Documento Visión, que contiene una visión general de los requerimientos del proyecto, las características claves y las restricciones principales de la aplicación Web.
- Estimar costos del proyecto: consiste en definir los costos aproximados de los recursos y las tareas necesarios para completar las actividades del proyecto. El Líder de la solución de software debe estimar el esfuerzo y el tiempo que se requieren para completar una actividad, para definir el costo total de la misma. El Líder de la solución de software considera las posibles variaciones del estimado final con el propósito de mejorar la administración del presupuesto del proyecto. El método de estimación de costos más conveniente para el proyecto depende del alcance del mismo, del nivel de precisión necesario y de las prácticas de la organización. Estas estimaciones son necesarias para establecer un presupuesto para el proyecto y para asignar un precio para el Portal Web. El Líder de la solución de software incluye los costos del proyecto en el Documento Visión.
- Establecer etapas de desarrollo: el Líder de la solución de software determina las etapas que se deben realizar para construir el software gradualmente. Se definen los objetivos de cada etapa y se establece el orden de las mismas. En estas etapas se describen todas las actividades que hay que realizar para obtener un conjunto de artefactos determinado y los elementos que contribuyen a la siguiente etapa. Se especifican los trabajadores que intervienen en cada etapa y las actividades que cada uno debe realizar. La planificación en fases del desarrollo del proyecto contribuye a que el equipo de desarrollo logre mayor organización y eficiencia en el trabajo durante el ciclo de vida

del proyecto. El Líder de la solución de software incluye las etapas de desarrollo del proyecto en el Proyecto Técnico.

- Definir cronogramas de ejecución: consiste en definir una lista de todos los elementos terminales del proyecto con sus fechas previstas de comienzo y final. El cronograma muestra las actividades del proyecto presentadas por etapas semanales y/o mensuales, según sea el caso hasta su conclusión. Para crear el cronograma del proyecto el Líder de la solución de software se basa en la estructura de descomposición del trabajo, en la estimación de esfuerzo para cada tarea y en la lista de los recursos con la disponibilidad de cada uno. El cronograma consiste en una estimación, cada fecha en él se estima, si los desarrolladores no concluyen las actividades en el tiempo establecido, el cronograma será inexacto.
- Preparar el ambiente de trabajo y recursos humanos: para determinar el ambiente de trabajo el Líder de la solución de software debe tener en cuenta un conjunto de condiciones bajo las cuales se va a realizar el trabajo, estas incluyen factores físicos, sociales, psicológicos y medioambientales. Estos factores influyen en la productividad de los trabajadores, pues los desarrolladores necesitan sentirse bien dentro de la empresa, que sus necesidades estén cubiertas, además de que prevalezca un clima laboral agradable. Para determinar las personas que se requieren para el desarrollo del Portal Web, el Líder de la solución de software debe seleccionar un conjunto de desarrolladores cuyas habilidades, motivación y características personales estén de acuerdo con la organización.
- Definir alcance del proyecto: consiste en establecer los límites de lo que exactamente se va a hacer en el proyecto. Cuanto más acotado esté el alcance habrá menos riesgo en el proyecto, pues se conocerá con mayor precisión lo que se va a realizar. El Líder de la solución de software incluye el alcance del proyecto en el Documento Visión.
- Definir objetivos del Portal Web: el Líder de la solución de software determina el propósito del Portal Web, cuáles son los resultados que el equipo de desarrollo debe obtener al concluir el ciclo del vida del proyecto. Es importante que los objetivos se definan de forma correcta, pues de ello depende que los recursos y esfuerzos destinados para darles cumplimiento se empleen eficientemente. El Líder de la solución de software incluye los objetivos del proyecto en el Proyecto Técnico.

- Definir la audiencia del Portal Web: consiste en determinar cuáles serán los grupos de usuarios que interactúan con el Portal Web. El Arquitecto de información es el encargado de agrupar los usuarios para un propósito común. Para reunir a los usuarios en un mismo grupo, estos deben tener características, privilegios y propósitos similares. El Líder de la solución de software incluye la audiencia del Portal Web en el Documento Visión.
- Definir la lista de riesgos del proyecto: consiste en determinar todas las posibles situaciones que pueden perjudicar el proyecto. El Líder de la solución de software elabora la lista de riesgos, considerando la visión del sistema para poder prever, ya desde el principio del proyecto, cuáles son los riesgos posibles. La lista de riesgo permite a los desarrolladores anticiparse a los riesgos y mitigarlos mediante estrategias definidas. Teniendo en cuenta que un riesgo puede disminuir la probabilidad de éxito en un proyecto o eliminarla por completo, el equipo de desarrollo debe estar informado sobre los posibles riesgos y dar respuesta ante la ocurrencia inminente de algún riesgo. La lista de riesgos generada se revisa constantemente para evaluar la efectividad de las estrategias de mitigación de riesgos, lo cual produce cambios en el plan del proyecto y las iteraciones subsecuentes.
- Impartir cursos de capacitación al equipo de desarrollo: el Líder de la solución de software se encarga de designar el personal calificado que impartirá los cursos de capacitación al equipo de desarrollo. La duración de los cursos de capacitación será en correspondencia con la cantidad de conocimiento que deben recibir los desarrolladores. La capacitación estará orientada a los roles del grupo de trabajo, cada rol recibirá una instrucción especializada dependiendo el trabajo que debe realizar dentro en el proyecto. El Líder de la solución de software realiza el Plan de Capacitación, que contiene las estrategias a seguir para capacitar a los trabajadores del proyecto.
- Definir la infraestructura tecnológica para el desarrollo del proyecto: consiste en diseñar e implantar una infraestructura tecnológica de hardware y software que provea una plataforma segura y estable para los desarrolladores. El Líder de la solución de software determina según el personal del equipo de desarrollo y las necesidades del proyecto, los dispositivos de hardware y las herramientas que se deben utilizar para realizar el proyecto.

Artefactos de la fase de Inicio:

- Contrato: En el Contrato acuerdan ambas partes (cliente y equipo de desarrollo) trabajar en conjunto por el desarrollo del Portal Web y especifican qué tareas le corresponde a cada parte.
- Proyecto Técnico: El Proyecto Técnico recoge las definiciones con el cliente en cuanto a fases de desarrollo, entregables y documentos rectores, hitos de cada fase, roles del desarrollo del Portal Web e incluso la propuesta de solución con un alcance bien delimitado. Incluye, además, la infraestructura tecnológica necesaria para el ambiente real con la arquitectura de software propuesta, el proceso de transferencia tecnológica y todo lo que concierne a instalación, montaje, publicación y soporte del Portal Web.
- Documento Visión: En el Documento Visión se incluye una visión general del producto, los beneficios y características que el producto debe tener, los costos de los recursos y las tareas del proyecto, el alcance y la audiencia de la aplicación Web y los posibles problemas que pueden ser resueltos con el proyecto.
- Cronograma del Proyecto: El Cronograma del Proyecto es una lista de todos los elementos terminales del proyecto con sus fechas previstas de comienzo y final.
- Lista de Riesgos: En la Lista de Riesgos se recoge la documentación de riesgos potenciales, sus magnitudes, y estrategias de mitigación.
- Plan de Capacitación: En el Plan de Capacitación se incluyen las estrategias que debe seguir el capacitador para realizar la capacitación de los trabajadores del proyecto. Se incluye además la justificación de la necesidad de la capacitación, así como el alcance, el presupuesto y el cronograma de la capacitación.

En la figura 6 se muestra la secuencia de las principales actividades a realizar en la fase de Inicio con los artefactos que generan y los roles responsables de realizar las actividades. Conjuntamente con el Líder de la solución de software debe trabajar el Psicólogo Organizacional en esta fase de Inicio, de manera que se esclarezcan los procesos organizacionales y se alcance un mayor entendimiento entre equipo de desarrollo y cliente. El análisis de los riesgos se especifica en la fase de Inicio pero continúa durante todo el desarrollo del proyecto.

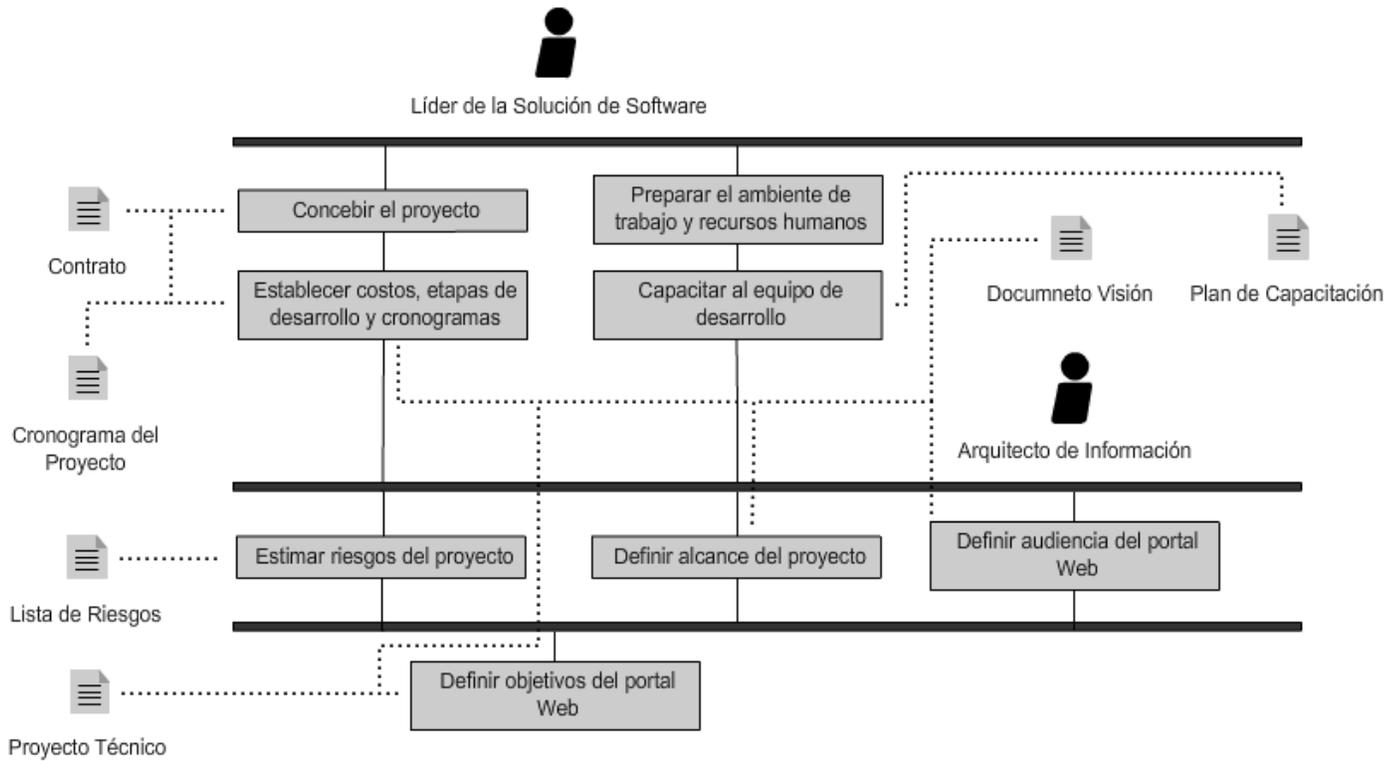


Figura 6: Representación de las principales actividades, artefactos y roles de la fase de Inicio.

2.3.2 Fase 2: Definiciones para el desarrollo

Las definiciones para el desarrollo involucran los elementos necesarios para proceder a la implementación del Portal Web. Las definiciones de esta fase involucran un fuerte trabajo en conjunto del equipo de desarrollo y el cliente.

Objetivos de la fase Definiciones para el desarrollo:

- Definición de la línea base de la arquitectura.
- Definición de los prototipos y las pautas del diseño gráfico del Portal Web.
- Levantamiento de requisitos e información para el desarrollo del Portal Web.

Trabajadores involucrados en la fase de Definiciones para el desarrollo:

- Arquitecto de información: es el encargado de estudiar los Portales Web homólogos al que se desea realizar, así como de definir las entidades de recursos de información y estructura del Portal Web.
- Diseñador gráfico: es el encargado de realizar los bocetos del diseño gráfico.
- Analista: es el encargado de definir los requisitos del Portal Web.
- Líder de la solución de software: es el encargado de definir la línea base de la arquitectura.

Actividades de la fase de Definiciones para el desarrollo:

- Estudiar sitios Web homólogos al que se pretende desarrollar: el Arquitecto de información realiza un análisis de los sitios similares al que se desea desarrollar y determina que elementos pueden servir de apoyo para realizar el Portal Web; analiza cuales son las tendencias mundiales para desarrollar Portales Web similares e incluye los análisis realizados en el Diagnóstico del Estudio de Sitios Web.
- Definir las entidades de recursos de información: consiste en definir cuáles son los contenidos y servicios que el Portal Web debe brindar, así como las fuentes de información. Teniendo en cuenta las funcionalidades que debe tener el Portal Web, el Arquitecto de información determina la información que va a estar disponible en las páginas, incluyendo texto, imágenes o cualquier otro tipo de información; además, determina que servicios adicionales debe ofrecer el sitio Web a los usuarios. Es importante definir cuáles serán los documentos, actividades o testimonios que darán origen a la información que se muestre en el Portal Web. El Arquitecto de información incluye las entidades de recurso de información en el Documento de Arquitectura de Información.
- Definir la estructura del Portal Web: tras haber determinado la información que se desea mostrar, el Arquitecto de información define la organización de la información a mostrar en cada una de las páginas. Determina las páginas que debe tener el sitio Web y organiza los contenidos y servicios, priorizando aquellos que tengan mayor importancia, en cada una de las páginas. El Arquitecto de información incluye la estructura del Portal Web en el Documento de Arquitectura de Información.
- Definir los requisitos del Portal Web: consiste en definir las condiciones o capacidades que tiene que mostrar el Portal Web, para que los usuarios puedan resolver un problema o lograr un objetivo. El Analista entrevista al cliente y especifica el comportamiento de entrada y salida del sistema, así como las acciones que el sistema debe ser capaz de realizar; define que información se va a

manejar en el sistema, que roles van a interactuar en el sistema y las relaciones que pueden ocurrir entre ellos, como se muestra la información y como se podrá navegar en el sistema; además identifica las restricciones que se imponen y las propiedades o cualidades que el Portal Web debe tener para cubrir las funcionalidades del mismo, estas son las características que hacen al producto atractivo, usable, rápido o confiable. Estos requerimientos del sitio Web deben ser especificados por escrito y posibles de probar o verificar; deben describir lo que el sistema debe hacer lo más abstracto y conciso posible, para evitar malas interpretaciones. El Analista realiza el documento Especificaciones de Casos de Usos, donde describe las funcionalidades del Portal Web en forma de casos de uso¹⁵.

- Realizar los bocetos del diseño gráfico del Portal Web: el Diseñador gráfico realiza un prototipo de interfaz para el Portal Web. Basándose en las funcionalidades del sistema realiza el diseño de los componentes de las páginas del Portal Web que el usuario visualiza; realiza el documento Manual de Pautas de Diseño Gráfico, donde se incluyen las características visuales de las páginas del Portal Web. El Diseñador gráfico realiza el Boceto del Portal Web, que consiste en un archivo psd que contiene la interfaz visual de la aplicación Web.
- Definir una línea base de arquitectura del Portal Web: consiste en definir la estructura de los componentes más significativos del Portal Web. El Líder de la solución de software determina cuáles son los casos de usos que describen las funcionalidades más importantes y críticas o algún requisito que debe priorizarse; decide cuales son los componentes de la arquitectura, cómo se relacionan entre ellos y el ambiente, y los patrones que restringen el diseño y la evolución del software. El Líder de la solución de software incluye la línea base de la arquitectura en el Documento de Arquitectura de Software.

Artefactos de la fase de Definiciones para el desarrollo:

- Documento de Arquitectura de Información: El Documento de Arquitectura de Información recoge la información a mostrarse en el Portal Web. Se definen las Entidades de Recursos de Información (ERI) y la estructura de las páginas del Portal Web.
- Especificaciones de Casos de Uso: En la Especificación de Casos de Uso se recogen los casos de uso del Portal Web así como las descripciones detalladas de dichos casos de uso y el Modelo de

¹⁵ Fragmentos de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores .

Casos de Uso. Todo el documento debe estar escrito en un lenguaje flexible tanto para el equipo de desarrollo como para los clientes.

- **Manual de Pautas de Diseño Gráfico:** En el documento se pautan aspectos del Portal Web como: dimensiones; pautas cromáticas y tipográficas; estilo de los menús, botones, imágenes, íconos, tablas, pie de página; el diseño de los banners, entre otros aspectos que deban ser definidos por un diseñador gráfico. Todos estos elementos se recogen en prototipos de pantalla dando una idea real del Portal Web.
- **Boceto del Portal Web:** El Boceto del Portal Web es un archivo psd¹⁶ que contiene el diseño gráfico del Portal Web. Se incluye el diseño de todas las pantallas de la aplicación Web.
- **Diagnóstico del Estudio de Sitios Web:** En el Diagnóstico del Estudio de Sitios Web se hace un estudio de sitios Web homólogos al que se pretende realizar. En el documento se identifican los servicios e información que brindan los sitios y se analiza la estructura de información y el diseño y forma de presentar los contenidos. Ver **Anexo 1**
- **Documento de Arquitectura de Software:** En el documento se recogen los requerimientos y objetivos del software que tienen un impacto significativo en la arquitectura, así como los casos de uso arquitectónicamente significativos y otros elementos significativos de la aplicación Web.

En la figura 7 se representan las principales actividades de la fase Definiciones para el desarrollo con los artefactos generados y los roles responsables de las actividades.

¹⁶ Extensión de los ficheros creados por Adobe Photoshop.

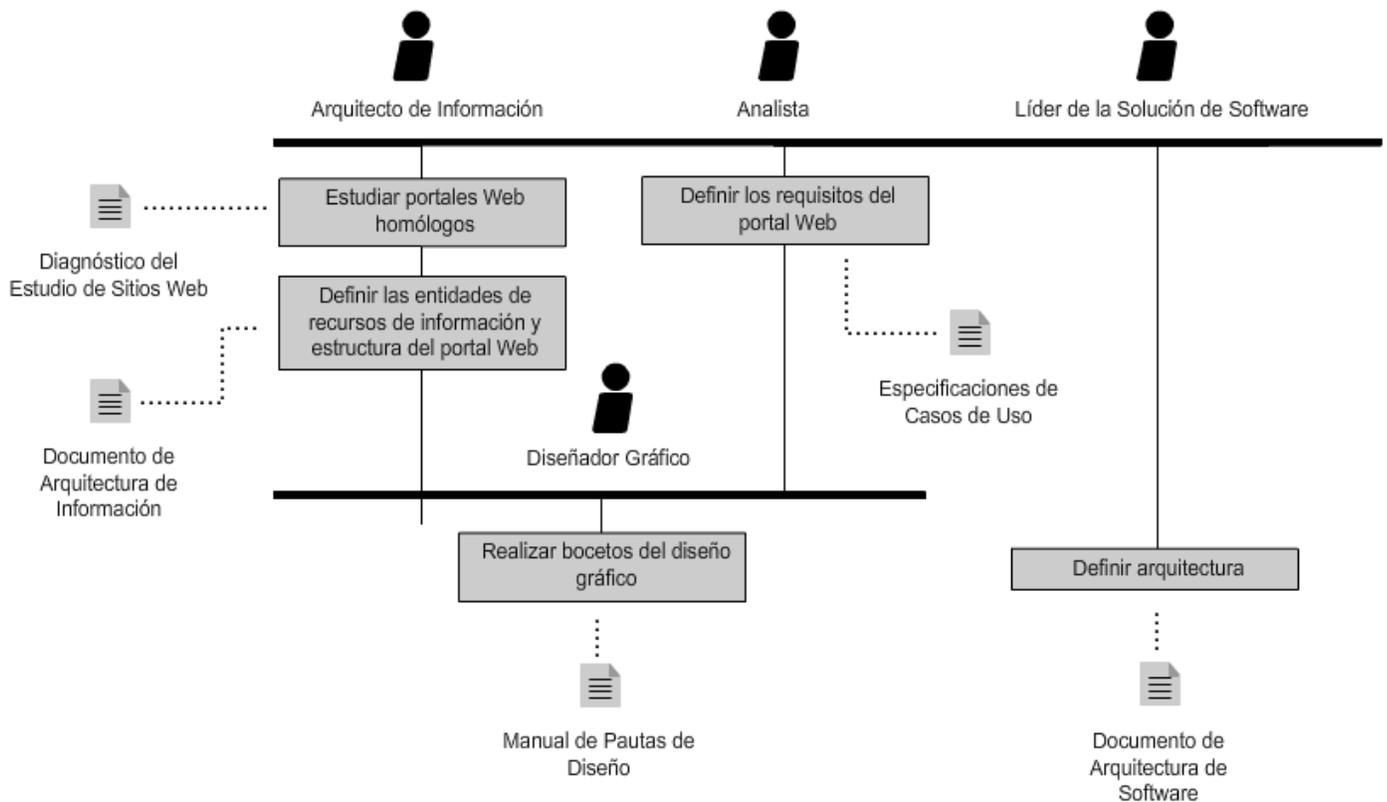


Figura 7: Representación de las principales actividades, roles y artefactos de la fase Definiciones para del desarrollo.

2.3.3 Fase 3: Diseño, implementación y pruebas

En la fase de Diseño, implementación y pruebas se lleva a cabo el diseño e implementación de los requerimientos del Portal Web, de igual forma se realizan pruebas al software que favorecen la calidad del producto y la satisfacción del cliente.

Objetivos de la fase de Diseño, implementación y pruebas:

- Realización del análisis, diseño e implementación del producto.
- Realización de las pruebas del Portal Web.
- Construcción de la documentación de cara al usuario.
- Preparación del ambiente y personal para desplegar el Portal Web.

Trabajadores involucrados en la fase de Diseño, implementación y pruebas:

- Analista: es el encargado de realizar el análisis de los requisitos y construir los manuales de uso y de acceso al Portal Web.
- Desarrolladores: son los encargados de estudiar los módulos de Drupal existentes y realizar el diseño de los requisitos, implementar los módulos necesarios y el tema, e integrarlos al CMS Drupal.
- Equipo de Calidad: es el encargado de realizarle las pruebas de calidad al software.
- Administrador del Portal Web: es el encargado de configurar el CMS Drupal.

Actividades de la fase de Diseño, implementación y pruebas:

- Estudiar los módulos de Drupal existentes: los Desarrolladores analizan los módulos existentes para definir cuáles podrían ayudar en la implementación del Portal Web. Se determina si los módulos implementados se pueden utilizar íntegramente o si los desarrolladores necesitan modificar algunos elementos de los módulos para cumplir con los requerimientos de la aplicación Web.
- Realizar el análisis del Portal Web: consiste en obtener una visión del sistema que se preocupa de ver qué se hace en el Portal Web sin tener en cuenta los elementos de la implementación. Se identifican las clases del análisis en: clases de interfaz, clases de control y clases de entidad. Se confecciona un diagrama de clases del análisis y un diagrama de colaboración por cada caso de uso del sistema. El Analista realiza el Modelo de Análisis.
 - Clase interfaz: clase que modela la interacción del actor con el sistema, o sea una clase que representa la interacción actor-caso de uso, sin tener en cuenta que en la solución puede tener más de una interfaz dentro de cada caso de uso para cada usuario.
 - Clase de control: clase que coordina el trabajo de uno o unos pocos casos de uso, coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso. Define el flujo de control y las transacciones dentro de un caso de uso delegando el trabajo a otros objetos.
 - Clase entidad: clase que modela la información del sistema, información que posee larga vida y a menudo es persistente; la información puede tratarse de fenómenos, conceptos o sucesos que ocurren en el mundo real.

- Diagrama de clases del análisis: es un artefacto en el que se representan los conceptos en un dominio del problema. Representa los elementos del mundo real, sin tener en cuenta la implementación automatizada de estos elementos.
- Diagrama de colaboración: muestra las interacciones de los objetos organizadas en torno a los objetos y las relaciones que ocurren entre ellos; explica gráficamente las interacciones entre las instancias del modelo, destacando la organización de los objetos que participan en la interacción.
- Realizar el diseño del Portal Web: es un refinamiento del análisis, consiste en definir cómo cumple el sistema sus objetivos. El diseño debe ser lo suficientemente explícito para que el Portal Web pueda ser implementado sin ambigüedades; de hecho, cuando la precisión del diseño es muy grande, la implementación puede ser completada por un generador de código automático. Se identifican las clases del diseño y se confecciona el diagrama de clases del diseño. Se confeccionan los diagramas de secuencia del diseño para cada evento de cada caso de uso. Se construye el diagrama de despliegue y se identifican los paquetes y subsistemas del diseño. Se identifican y se aplican los patrones de asignación de responsabilidades. Los Desarrolladores realizan el Modelo de Diseño.
 - Clases del diseño: Se especifican utilizando la sintaxis del lenguaje de programación elegido. Se especifica la visibilidad de los atributos y operaciones (public, protected, private); los métodos especificados corresponden a los métodos de la implantación.
 - Diagrama de clases del diseño: Es una representación más concreta que el diagrama de clases del análisis, teniendo en cuenta los elementos correspondientes a la implementación del Portal Web. Representa la parte estática del sistema y las clases del sistema y sus relaciones.
 - Diagrama de secuencia del sistema: muestra las interacciones entre objetos, ordenadas en secuencia temporal durante un escenario concreto. Si los casos de uso presentan varios eventos distintos, suele se debe crear un diagrama de secuencia para cada uno de ellos.
 - Subsistema del diseño: Es una forma de organizar los artefactos del modelo de diseño en piezas más manejables. Puede contener clases del diseño, realizaciones de casos de uso, interfaces y otros subsistemas. Pueden proporcionar interfaces que presentan la funcionalidad que exportan en términos de operaciones.

- Paquetes de diseño: es una colección de clases, relaciones, realizaciones de casos de uso, diagramas y otros paquetes que estén de alguna forma relacionados. Es usado para estructurar el modelo de diseño dividiéndolo en partes más pequeñas. A diferencia de los subsistemas de diseño, los paquetes no ofrecen una interfaz formal.
 - Diagrama de despliegue: muestra cómo y dónde se desplegará el sistema. Las máquinas físicas y los procesadores se representan como nodos. Además muestran la configuración en funcionamiento del sistema incluyendo su software y su hardware. Para cada componente de un diagrama es necesario que se deba documentar las características técnicas requeridas.
- Instalar el CMS Drupal: el Administrador del Portal Web instala Drupal en un ambiente de prueba y realiza una configuración temporal hasta que se concluya el tema del Portal Web. El Diseñador de base de datos crea la base de datos del Portal Web y le realiza una copia archivándola en el Script de la Base de Datos del Portal Web.
 - Implementar el tema del Portal Web: para crear un nuevo tema se recomienda encontrar un tema implementado que presente características similares al que se desea implementar y realizarle modificaciones hasta lograr el tema que se ideó para el Portal Web. El Diseñador gráfico implementa el aspecto visual del Portal Web, incluyendo entre otros muchos elementos, los colores, los tipos de letra y la ubicación del contenido mostrado en las páginas. El tema es un fichero que se encuentra en el directorio *themes* del servidor e incluye plantillas con la estructura del sitio Web, hojas de estilo e imágenes.
 - Implementar los módulos del Portal Web: consiste en implementar los módulos necesarios para poder realizar algunas funcionalidades específicas que requiere el Portal Web. Actualmente existe un número realmente grande de módulos disponibles, de temáticas diversas y de gran calidad. Para crear nuevos módulos se recomienda reutilizar módulos existentes que realicen funcionalidades semejantes a las que debe tener el Portal Web. Al concluir los módulos se obtienen ficheros con extensión *.module* que contienen funciones escritas en PHP; estos módulos se instalan simplemente copiándolos en el directorio *modules* del servidor, y aunque generalmente se componen solamente del fichero *.module*, a veces vienen acompañados de otros archivos complementarios como hojas de estilo CSS, entre otros.

- Configurar el CMS Drupal: el Administrador del Portal Web monta el tema del Portal Web e incluye los módulos creados para cumplir los requerimientos de la aplicación Web, obteniendo una versión funcional del sistema. Se habilitan los módulos insertados para que se puedan realizar las funciones que estos permiten. Se definen los permisos de usuarios y se restringen las funcionalidades que cada grupo de usuarios puede realizar.
- Realizar pruebas al producto: el Equipo de calidad realiza diferentes pruebas a la aplicación para comprobar diferentes aspectos que definen la calidad del producto. Inicialmente se prueban los módulos con el objetivo de verificar que estos permiten realizar las actividades para las que fue desarrollado. Se comprueba que el sitio realizado cumpla con los requisitos de diseño especificados por el cliente; se comprueba además que no existan errores ortográficos en el diseño, que las vistas de los contenidos sean específicamente como fueron previamente diseñadas. Se realizan pruebas de liga, o sea, se comprueba que cada hipervínculo vaya a la sección que le corresponde y muestre el contenido correcto. Al realizar las pruebas al sitio se verificará que exista homogenización en el idioma escogido y que cada actor entre al Portal de acuerdo a sus privilegios. Se introducen datos erróneos para que el sistema falle y poder revelar sus debilidades. Se verifica si el Portal Web acepta más de un usuario con el mismo alias. Se verifican cuántas conexiones simultáneas se pueden hacer sin causar errores en la aplicación y cuántas peticiones puede manejar el servidor sin colgarse. Se hacen pruebas de tiempo de respuesta para calcular el tiempo que se demora la aplicación para realizar consultas. Se realizan pruebas al Portal Web en general, para comprobar que el sistema funciona correctamente en conjunto; se hacen con el objetivo de buscar diferencias entre los requerimientos y el funcionamiento del Portal Web.
- Construir los manuales de uso y acceso al Portal Web: consiste en definir los procedimientos para que los usuarios puedan acceder, administrar y gestionar los contenidos del Portal Web. Teniendo en cuenta los permisos de cada grupo de usuario se define como los usuarios se relacionan con los contenidos de la aplicación. El analista incluye todos estos procedimientos detallados en el Manual de Usuario.

Artefactos de la fase de Diseño, implementación y pruebas:

- Modelo de Análisis: El Modelo de Análisis contiene el diagrama de clases del análisis y los diagramas de colaboración por cada caso de uso del sistema.
- Modelo de Diseño: El Modelo de Diseño contiene el diagrama de clases del diseño, los diagramas de secuencia del diseño para cada evento de cada caso de uso, el diagrama de despliegue y se identifican los paquetes y subsistemas del diseño.
- *Script* de la Base de Datos del Portal Web: El *Script* de la Base de Datos contiene una copia actualizada de la base de datos del Portal Web.
- Aplicación Web: La Aplicación Web es una carpeta donde se recogen los módulos, tema y archivos necesarios para implantar el Portal Web.
- Manual de Usuario: El Manual de Usuario es una guía bien detallada de cómo acceder, administrar y gestionar los contenidos del Portal Web.

En la siguiente figura se muestra las principales actividades, roles y artefactos de la fase Diseño, implementación y pruebas. En la figura queda reflejada la realización de pruebas al software por el equipo de calidad, pero es importante señalar que el equipo de calidad está presente durante todo el ciclo de vida del proyecto garantizando la calidad de todo el producto.

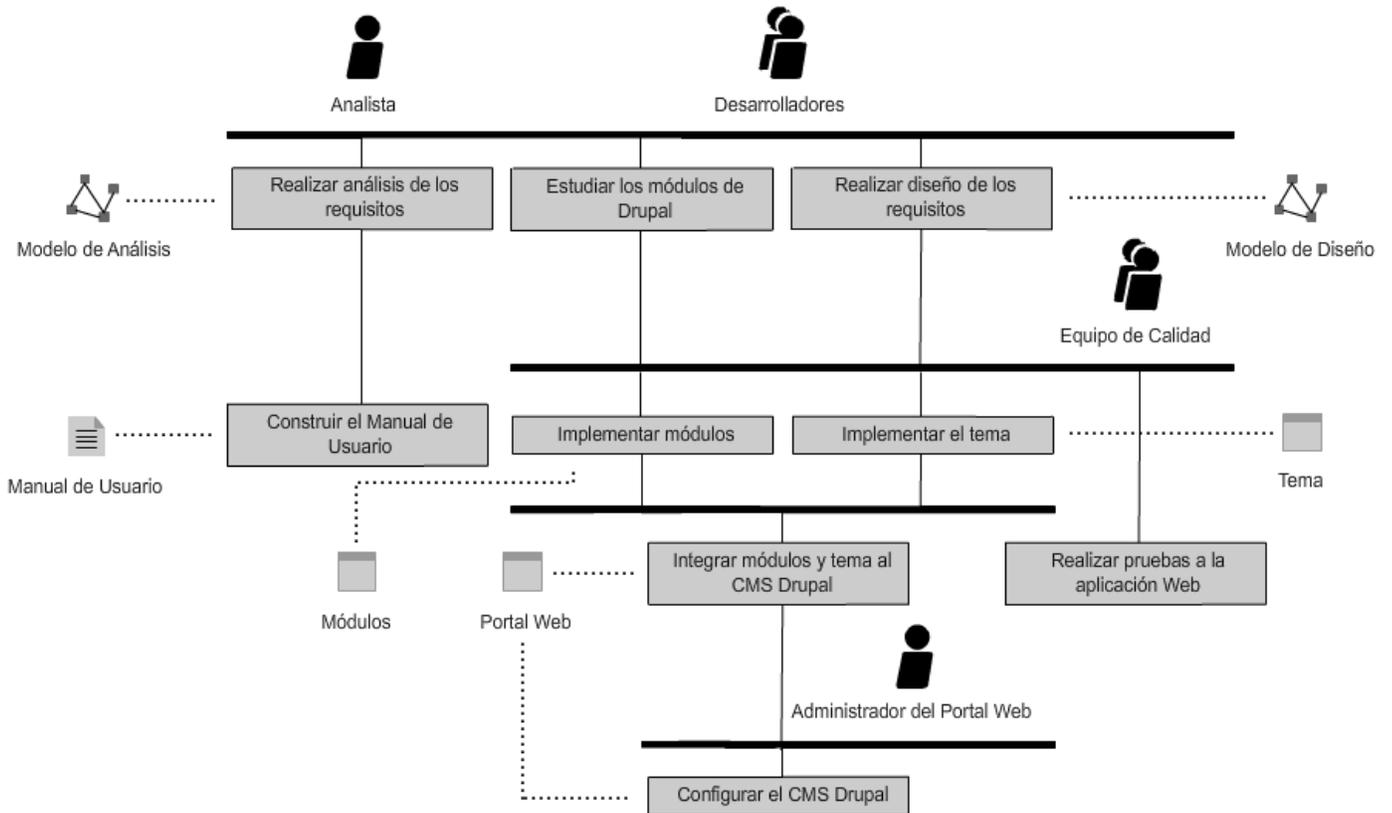


Figura 8: Representación de las principales actividades, roles y artefactos de la fase Diseño, implementación y pruebas.

2.3.4 Fase 4: Implantación y publicación

La fase de implantación consiste en instalar el Portal Web en un ambiente real de manera que el cliente pueda realizar el montaje de los contenidos. Se realiza, además, las pruebas de aceptación del cliente y las pruebas de conectividad. Una vez implantado el Portal Web se hace público a la comunidad de usuarios a la que va dirigido.

Objetivos de la fase de Implantación y publicación:

- Instalación y configuración del Portal Web en el ambiente real.
- Realización de las pruebas de aceptación del cliente.
- Publicación del Portal Web.
- Capacitación de los usuarios finales (administradores del Portal Web).

Trabajadores involucrados en la fase de Implantación y publicación:

- Equipo de despliegue: es el encargado de instalar el Portal Web en el ambiente real y de configurar el Portal Web.
- Administrador del Portal Web: es el encargado de montar los contenidos reales en el Portal Web y de publicar el Portal Web.
- Equipo de calidad: es el encargado de realizar las pruebas de aceptación junto al cliente.
- Capacitador: es el encargado de capacitar a los usuarios finales del Portal Web.

Actividades de la fase de Implantación y publicación:

- Instalar el Portal Web: el Equipo de despliegue instala la aplicación Web terminada por los desarrolladores en el establecimiento donde se determinó con el cliente que se instalaría el producto final. Antes de instalar el Portal Web se verifica que el ambiente real donde se instalara el software cumple con las normas establecidas para que la aplicación Web se instale sin problemas. Se instala la base de datos utilizando el Script de la Base de Datos en el servidor seleccionado por el cliente para la base de datos. El Equipo de despliegue construye el Manual de Instalación, donde se describe detalladamente los pasos a seguir para realizar una correcta instalación del Portal Web.
- Configurar el Portal Web y los servidores del producto: el Equipo de despliegue configura los módulos para que se puedan realizar las funciones que estos permiten. Se realizan las configuraciones finales en el Portal Web y en la base de datos, en el ambiente real, para que la aplicación funcione plenamente.
- Realizar las pruebas de aceptación con el cliente: después de haber instalado y configurado el Portal Web, el Equipo de calidad y el cliente realizan pruebas al producto para determinar si el producto cumple con todos los requerimientos establecidos. Se identifican las no conformidades que fueron detectadas por el cliente y se erradican. El Equipo de calidad realiza el Acta de Aceptación, donde se determina por ambas partes que el producto realizado está listo para ser entregado al cliente y finalmente publicado.
- Montar los contenidos del Portal Web: consiste en introducir los datos reales que finalmente aparecerán publicados en el Portal Web. Generalmente es el cliente quien posee la información que se publicará en el sitio Web, esta información es analizada y corregida por el Administrador del

Portal Web, luego de verificar que los contenidos aparecen en el formato correcto, se aprueba su publicación en el Portal Web.

- Impartir cursos a los usuarios del Portal Web: consiste en capacitar a los administradores del Portal Web definidos por el cliente, para que estos puedan tomar el control de la aplicación Web. El Capacitador organiza un programa de capacitación que permita que los administradores del sistema puedan prescindir del equipo de desarrollo para gestionar la aplicación; aunque el equipo de desarrollo le dará soporte a la aplicación en caso de surgir algún inconveniente con el producto.
- Publicar el Portal Web: después que el equipo de desarrollo y el cliente acuerdan que el producto está listo para ser publicado, el Administrador del Portal Web publica la aplicación Web y finalmente el producto queda en manos del cliente cumpliendo con todos los requisitos definidos.

Artefactos de la fase de Implantación y publicación:

- Manual de Instalación: en el documento Manual de Instalación se recogen todos los aspectos a tener en cuenta para poner en práctica el Portal Web. Es un compendio de todas las acciones y configuraciones que deben hacerse en los servidores para la exitosa publicación del Portal Web.
- Acta de Aceptación del Producto: es el documento donde ambas partes (cliente y equipo de desarrollo) concuerdan en la aceptación del producto por parte del cliente.

En la siguiente figura se muestran las principales actividades, roles y artefactos generados en la fase Implantación y publicación.

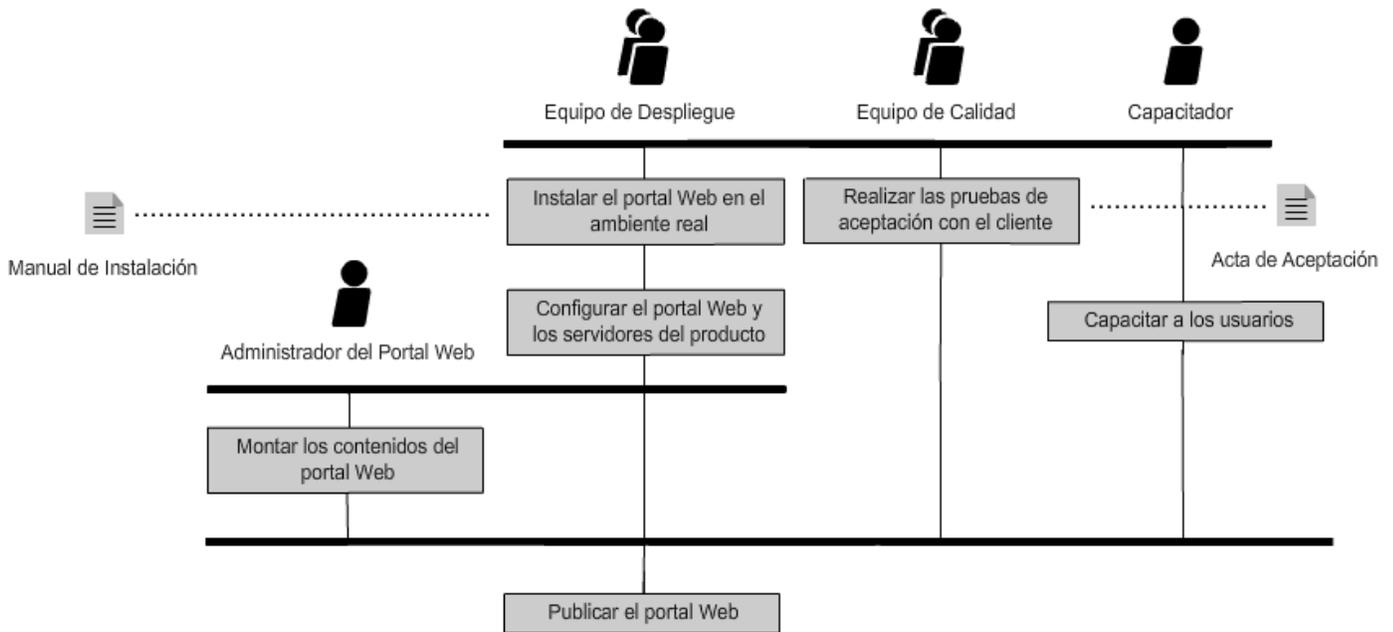


Figura 9: Representación de las principales actividades, roles y artefactos de la fase Implantación y publicación.

2.4 Conclusiones

Después de haber definido los elementos necesarios para que el procedimiento de desarrollo sea exitoso, se puede concluir que el desarrollo de Portales Web basados en el CMS Drupal está dividido en cuatro fases y en cada una de ellas se realizan actividades adecuadas al CMS y son generados artefactos que permiten el correcto desarrollo del producto. También se puede destacar que el desarrollo de Portales Web, de manera general, requiere de un equipo de desarrollo dividido en roles con funcionalidades específicas. El procedimiento realizado abarca todas las etapas de construcción de un Portal Web, comenzando desde la planificación del proyecto hasta la implantación y publicación del producto final; para que el equipo de desarrollo realice la aplicación Web dentro de los términos establecidos y cumpla las expectativas del cliente.

CAPÍTULO 3: VALIDACIÓN DEL PROCEDIMIENTO PROPUESTO

3.1 Introducción

Este capítulo se centra en un proceso de validación del procedimiento de trabajo propuesto. Para aceptar y validar el procedimiento se utiliza el método Delphy, que es una técnica de investigación social que tiene como objeto la obtención de una opinión grupal a partir de un grupo de expertos. El panel de expertos está compuesto por varios jefes de proyectos y otras personas que tienen experiencia trabajando con el CMS Drupal. Los expertos seleccionados respondieron un cuestionario donde se miden todos los aspectos necesarios para determinar si el procedimiento propuesto puede ser aplicado en proyectos reales.

3.2 Método de evaluación

Se utiliza como método de evaluación el método Delphy, debido a que es considerado como uno de los métodos subjetivos de pronosticación más confiables. Constituye un procedimiento para confeccionar un cuadro en la evaluación de situaciones complejas, a través de la elaboración estadística de las opiniones de expertos en el tema tratado. Permite rebasar el marco de las condicionantes actuales más señaladas de un fenómeno y alcanzar una imagen integral y más amplia de su posible evolución, reflejando las valoraciones individuales de los expertos, las cuales podrán estar fundamentadas, tanto en un análisis estrictamente lógico como en su experiencia intuitiva.

“... el Delphy es la utilización matemática del juicio intuitivo de un grupo de expertos para obtener un consenso de opiniones informadas” (Criterio de Expertos: Método Delphy. 2006)

El método Delphy fue creado con el objetivo de elaborar pronósticos a largo plazo, referentes a posibles acontecimientos en varias ramas de la ciencia, la técnica y la política.

La esencia de este método consiste en la organización de un diálogo anónimo entre los expertos consultados individualmente, mediante cuestionarios, con vistas a obtener un consenso general o, al menos, los motivos de la discrepancia, la confrontación de las opiniones se lleva a cabo mediante una serie de interrogantes sucesivas, entre cada una de las cuales la información obtenida sufre un procesamiento estadístico - matemático.

Las principales características del método son las siguientes:

- Anonimato: se expresa a través del no-conocimiento de las respuestas, puesto que los miembros del grupo contestan las preguntas sin confrontarse, incluso sin conocerse entre sí.

- Retroalimentación controlada: después de cada ronda de preguntas se tabulan las respuestas y se procesan de forma tal, que antes de la siguiente ronda los participantes pueden evaluar los resultados de la ronda anterior, así como las razones dadas para cada respuesta y su dispersión del promedio.
- Respuesta estadística del grupo: entre cada ronda de preguntas, la información obtenida se procesa por medio de técnicas estadístico-matemáticas, como, por ejemplo, el diseño experimental no paramétrico, las que dotan al investigador de un instrumento objetivo y concreto en el cual pueden apoyarse para tomar una decisión final.

Los elementos fundamentales a tener en cuenta para poner en práctica el método son la selección de los expertos y la elaboración del cuestionario. A continuación se evidencia como fueron desarrollados estos dos aspectos en el trabajo.

3.3 Proceso de selección de expertos

“Se entiende por experto, tanto al individuo en sí como a un grupo de personas u organizaciones capaces de ofrecer valoraciones conclusivas de un problema en cuestión y hacer recomendaciones respecto a sus momentos fundamentales con un máximo de competencia.” (Criterio de Expertos: Método Delphy. '06)

En la planificación del criterio de expertos se parte de la concepción inicial del problema y la selección de los expertos como pasos previos fundamentales para la aplicación del criterio.

Para la selección del grupo de experto se realizaron las siguientes actividades:

- Determinación de las áreas del conocimiento que deben dominar los expertos: Partiendo del problema planteado en la Introducción se determinó que los expertos a consultar debían dominar las siguientes áreas del conocimiento: metodologías de desarrollo de software para realizar aplicaciones Web, trabajo con el CMS Drupal, ingeniería de software y modelado de procesos. Los expertos seleccionados deben ser un grupo de personas que puedan dar un criterio concluyente sobre estas materias y emitir valoraciones importantes con un alto nivel de conocimiento.
- Elaboración del listado de expertos candidatos: Luego de determinar las áreas del conocimiento se elaboró un listado de expertos candidatos; se seleccionaron personas que están vinculadas a la producción de software en la Universidad y cuentan con los conocimientos necesarios para emitir una valoración. El listado inicial estaba conformado por ocho expertos, distribuidos de la siguiente manera: 6 especialistas con varios años de experiencia como líderes de proyecto que realizan

aplicaciones Web utilizando CMS Drupal y dos estudiantes de quinto año que desempeñan roles en un proyecto productivo que utiliza el CMS Drupal.

- Selección de expertos: Para seleccionar los expertos se tuvieron en cuenta las siguientes cualidades: competencia, creatividad, disposición a participar en la encuesta, conformismo, espíritu colectivista y auto crítico, capacidad de análisis, responsabilidad, sinceridad y seriedad. Estas características permiten que las opiniones recogidas sean confiables y válidas para valorar el procedimiento propuesto.

Las características de los expertos influyen decisivamente en la confiabilidad de los resultados obtenidos. Estas características son: calificación técnica, capacidad de emitir una decisión al respecto, conocimientos específicos sobre el tema a evaluar, disposición a participar, entre otros. (Febles '03) (Criterio de Expertos: Método Delphy. '06)

De los expertos candidatos escogidos, cinco estuvieron dispuestos a participar en el proceso de validación del procedimiento. La siguiente tabla muestra los datos de los expertos seleccionados para realizar la encuesta.

Expertos	Años vinculados a proyectos en la UCI	Graduado de	Rol que ocupa en el proyecto
Experto 1	5	Ingeniero informático	Líder de proyecto
Experto 2	4	Ingeniero informático	Líder de proyecto
Experto 3	4	Ingeniero informático	Líder de proyecto
Experto 4	2	Estudiante 5to año	Diseñador BD
Experto 5	2	Estudiante 5to año	Programador

Figura 10: Datos de los integrantes del Panel de Expertos.

Una vez confeccionado el panel de expertos se invitó a cada uno de ellos de forma personal para que participaran en el proceso de validación y aceptación de la propuesta. Se le detalló de una forma clara y precisa los objetivos de este proceso explicando en qué consistía el trabajo en general, la propuesta a evaluar, la realización del cuestionario, así como el plazo de entrega.

3.4 Elaboración del cuestionario

La elaboración del cuestionario tiene en cuenta los principios de la teoría de la comunicación y crea mecanismos que reducen los sesgos en las respuestas. Se facilita que el experto valore alternativas a sus respuestas y se solicita que exponga sus argumentos. Las preguntas realizadas tienen un enfoque investigativo y se centran principalmente en los principios básicos que debe cumplir la propuesta presentada, además de permitir que las respuestas fueran abiertas, y en todos los casos con posibilidad de emitir su criterio personal en cuanto a su experiencia en el proyecto al que pertenece.

Para el procesamiento y análisis de la información se valora desde el punto de vista cualitativo y cuantitativo respectivamente. El análisis cualitativo es fundamentalmente para las preguntas de tipo abiertas, se leen detalladamente cada una de las respuestas y se resumen los elementos más comunes y esenciales.

El cuestionario establece trece preguntas, que permitieron ver la posibilidad real de que pueda ser aplicado el procedimiento de trabajo elaborado en los proyectos productivos de la Universidad. Las preguntas abarcan de manera general todos los aspectos fundamentales del procedimiento. Los aspectos a tener en cuenta en las preguntas son:

- Fases de desarrollo que conforman el procedimiento de trabajo.
- Actividades, roles y artefactos definidos en cada una de las fases establecidas.
- Alcance del procedimiento de trabajo.
- Aplicabilidad en un ambiente real del procedimiento definido.

Se estableció una escala del uno al cinco, siendo uno el valor de menor importancia y cinco el de mayor. Estas preguntas proporcionan una mayor riqueza en las respuestas que son brindadas por los expertos debido a que se les dio la posibilidad que expresaran su opinión y dieran sugerencias al trabajo propuesto.

En este proceso los expertos recibieron de forma personal los cuestionarios y se les pidió que completaran esta tarea en un mínimo de tiempo para analizar las respuestas y permitirle que hicieran preguntas en caso que hubiesen surgido dudas referentes al cuestionario. Las siguientes preguntas son las que conforman el cuestionario enviado a los expertos:

- a) ___ Evalúe de manera general, las fases de desarrollo que componen el procedimiento de trabajo.
- b) ___ Evalúe la fase de Inicio.

- c) ___ Evalúe las actividades de trabajo que establecidas para realizar en la fase de Inicio.
- d) ___ Evalúe la fase de Definiciones para el desarrollo.
- e) ___ Evalúe las actividades de trabajo definidas en la fase de Definiciones para el desarrollo.
- f) ___ Evalúe la fase de Diseño, implementación y pruebas.
- g) ___ Evalúe las actividades de trabajo establecidas para la fase de Diseño, implementación y pruebas.
- h) ___ Evalúe la fase de Implantación y publicación.
- i) ___ Evalúe las actividades de trabajo definidas en la fase de Implantación y publicación.
- j) ___ Evalúe los roles definidos en el procedimiento de trabajo.
- k) ___ Evalúe los artefactos definidos en el procedimiento de trabajo.
- l) ___ Evalúe el alcance del procedimiento de trabajo; teniendo en cuenta, si abarca o no, todo el procedimiento de creación de un Portal Web utilizando el CMS Drupal.
- m) ___ Evalúe la aplicabilidad del procedimiento de trabajo.
- n) ___ Evalúe si las actividades definidas en cada fase abarcan los objetivos de cada fase.

Además de las preguntas se les pide a los expertos que incluyan en la encuesta sus observaciones sobre el procedimiento propuesto.

3.4 Resultados de la evaluación

La siguiente tabla muestra el valor asignado por cada uno de los expertos del panel a cada pregunta que conforma el cuestionario para validar el procedimiento de trabajo.

Expertos	a	b	c	d	e	f	g	h	i	j	k	l	m	n
Experto 1	5	4	3	5	5	5	4	5	5	5	4	5	4	4
Experto 2	5	5	4	5	5	4	4	5	4	5	4	5	5	4
Experto 3	5	4	4	4	4	5	3	5	5	4	5	4	4	4
Experto 4	5	5	4	4	5	5	4	5	5	4	4	5	5	4
Experto 5	5	5	4	5	5	4	4	4	5	5	4	5	5	5

Figura 11: Respuesta de los expertos a las preguntas del cuestionario.

Con los datos anteriores se procede a calcular el grado de concordancia de los expertos con respecto al conjunto de preguntas. Este cálculo se realiza utilizando para ello el coeficiente de concordancia de Kendall, para lo que se hizo necesario asignar un rango a cada evaluación dada por el experto i a las j preguntas, este rango varía de **1** a **5**, de forma tal que, el valor **5** le corresponde a la evaluación mayor, y el valor **1** a la menor de las efectuadas por el experto. Nótese que la evaluación mayor otorgada por el experto no tiene por qué coincidir con la puntuación mayor de la escala definida.

Si el experto emplea la misma evaluación para más de una pregunta, el rango será igual a la media aritmética de las posiciones que deben ser adjudicadas. La suma de rangos que se obtiene a partir de los valores ya definidos se expresa como:

$$S_j = \sum_{i=1}^m R_{ij} \quad (1)$$

La fórmula (1) refleja la suma de rangos correspondientes a la evaluación realizada por los expertos a la pregunta j , donde R_{ij} es el rango asociado a la evaluación del experto i de la pregunta j .

Este valor S_j puede ser utilizado para comparar la importancia de diferentes preguntas, de modo que un mayor valor significará una mayor importancia.

Se define también:

$$\bar{S} = \frac{m(n+1)}{2} \quad (2)$$

La fórmula (2) representa la media de la suma de los rangos de cada pregunta j , donde m es el número de expertos y n la cantidad de preguntas de que componen el cuestionario.

Con estos valores es posible entonces calcular el coeficiente de concordancia de Kendall como:

$$k = \frac{12 \sum_{j=1}^n (S_j - \bar{S})^2}{m^2(n^3 - n) - m \sum_{i=1}^m T_i} = \frac{12 \sum_{j=1}^n S_j^2 - n \bar{S}^2}{m^2(n^3 - n) - m \sum_{i=1}^m T_i} \quad (3)$$

Donde T_i es el resultado de los rangos iguales, llamados también ligas, que ofreció el experto i a las preguntas, calculado como sigue:

$$T_i = \frac{\sum_{l=1}^l (t^3 - t)}{12} \quad (4)$$

Donde l es el número de grupos con rangos iguales para el experto i y t el número de observaciones dentro de cada uno de los grupos para el experto i . Si todas las evaluaciones realizadas por el experto son diferentes, entonces $T_i = 0$.

Los valores del coeficiente de Kendall oscilan entre $0 < K < 1$, donde para $K = 1$ existe una concordancia total de criterios. Sin embargo un valor bajo de K significa una débil concordancia que, generalmente, es producida porque no existe realmente una alta concordancia o existen grupos entre los expertos con una elevada concordancia, pero con criterios opuestos.

Como se había señalado, el análisis del coeficiente de concordancia de Kendall puede evidenciar un débil consenso de opiniones entre los expertos, producido, entre otras cosas, por la existencia de grupos con criterio semejantes, pero opuestos entre sí.

Con el objetivo de comprobar la existencia de estos grupos, se puede llevar a cabo un análisis al nivel de cuestionario entre los diferentes expertos, lo que permite observar la concordancia de criterios a un nivel superior.

Después de haber obtenido el grado de concordancia entre los expertos, se procede a realizar la prueba de significación de hipótesis, planteándose la hipótesis nula y la alternativa de la siguiente forma:

Hipótesis nula

H₀: Los expertos no están de acuerdo con la propuesta del procedimiento de trabajo.

Hipótesis alternativa

H_1 : Los expertos están de acuerdo con la propuesta del procedimiento de trabajo.

Se determina X^2 , calculado como:

$$X^2_{cal} = m (n - 1) k \quad (5)$$

Por otro lado se busca el X^2 tabulado en la tabla del percentil de la distribución X^2 con un nivel de significación α y $n - 1$ grados de libertad, representada por:

$$X^2_{tab} = X^2_{\alpha;n-1} \quad (6)$$

Se comparan X^2_{cal} y X^2_{tab} , si se obtiene que $X^2_{cal} > X^2_{tab}$ entonces se rechaza H_0 y se infiere que los expertos si están de acuerdo con la propuesta, por lo que se considera válida la hipótesis alternativa H_1 .

3.4.1 Resultados obtenidos

A partir de los valores que les fueron otorgados a cada una de las preguntas del cuestionario por los expertos se aplica la fórmula (1), donde se calcula la suma de rangos correspondiente a las preguntas. La siguiente figura muestra los valores de las sumas de rangos.

Preguntas	a	b	c	d	e	f	g	h	i	j	k	l	m	n
Sj	25	22	18	23	23	22	18	24	24	23	20	24	23	21

Figura 12: Valores de la suma de rangos de las preguntas del cuestionario.

La comparación de los valores obtenidos para S_j conducen a que el mayor resultado obtenido corresponde a la pregunta **a**, esto refleja que el panel de expertos concuerda en que la definición de las fases del procedimiento de trabajo de manera general, es el aspecto más significativo del procedimiento propuesto; de esta forma, en orden descendiente se determina la calidad de cada aspecto según el criterio de los expertos.

Utilizando la fórmula (2), se procede a calcular la media de la suma de rangos, considerando que $m = 5$ y $n = 14$.

$$\bar{S} = \frac{5 * (14 + 1)}{2} = 37.5$$

Para calcular el coeficiente de concordancia de Kendall utilizando la fórmula (3), se realizan los siguientes cálculos parciales:

$$\sum S_j^2 = 625 + 484 + 324 + 529 + 529 + 484 + 324 + 576 + 576 + 529 + 400 + 576 + 529 + 441 = 6442$$

Utilizando la fórmula (4) se calcula el resultado de los rangos iguales.

$$\sum T_i = 52 + 59.5 + 52 + 59.5 + 70 = 293.5$$

Una vez obtenidos todos los valores necesarios, se calcula el coeficiente de concordancia de Kendall, utilizando la fórmula (3).

$$K = \frac{12 * 6442 - 13 * 1406.25}{25 * 2730 - 5 * 293.5} = 0.88$$

Los resultados arrojan que **K** tiene un valor entre $0 < K < 1$, donde **K = 0.88** se aproxima a **1**, por lo que se puede inferir que los expertos estuvieron generalmente de acuerdo en los criterios emitidos en la encuesta realizada.

Para calcular la prueba de significación de hipótesis se utiliza la fórmula (5).

$$X_{cal}^2 = 5 * (14 - 1) * 0.88 = 57.2$$

Para comparar el valor de X_{cal}^2 y X_{tab}^2 , se busca en la tabla del percentil de la distribución X^2 con un nivel de significación $\alpha = 0.05$ y 4 grados de libertad, y utilizando la fórmula (6) se obtiene:

$$X_{tab}^2 = X_{0.05, 4}^2 = 9.4877$$

Siendo de $X_{cal}^2 = 57.2$ y $X_{tab}^2 = 9.4877$, donde $X_{cal}^2 > X_{tab}^2$, se rechaza la hipótesis nula H_0 , por lo que no es cierto que los expertos rechazan la propuesta del procedimiento de trabajo realizado; esto da lugar a que se considere válida la hipótesis alternativa H_1 , por lo que se puede afirmar que los expertos aprueban la propuesta del procedimiento de trabajo.

3.5 Conclusiones

Se han realizado todos los pasos que componen el proceso de validación del procedimiento de trabajo, desde la selección del panel de expertos y la elaboración del cuestionario hasta el empleo del método

Delphy para determinar si la propuesta es válida o no. Los resultados obtenidos de las respuestas dadas por los expertos demuestran que el panel de expertos, de manera general, considera que el procedimiento de trabajo realizado es válido. Esto infiere que el procedimiento de trabajo realizado puede ser utilizado por equipos de desarrollo que necesiten realizar aplicaciones Web utilizando el CMS Drupal.

CONCLUSIONES

Con la realización de la presente investigación se arribaron a las siguientes conclusiones:

- La necesidad de gestionar y difundir la información ha influido en el auge del desarrollo de los Portales Web, pues estos son una solución factible a la necesidad de proporcionar servicios informativos e interactivos de manera instantánea.
- Debido a la complejidad que de las aplicaciones Web fue necesario crear los Sistemas de Gestión de Contenidos, permitiendo que fuese más fácil crear, mantener, administrar y publicar sitios Web. Drupal es un típico CMS que permite gestionar fácilmente Portales Web.
- El CMS Drupal es una tendencia actual en el desarrollo de Portales Web debido a que es un sistema de gestión de contenidos de código abierto. Es multiusuario, multiplataforma, multilinguaje, extensible y modular.
- A partir del análisis de varias metodologías de desarrollo se llega a una estandarización de un procedimiento de trabajo para el desarrollo de Portales Web basados en el CMS Drupal que incluye un ciclo de trabajo completo, un equipo de desarrollo dividido en roles, tareas establecidas en cada fase del ciclo de trabajo y documentación en cada fase.
- El procedimiento propuesto se sometió al criterio de un panel de expertos, resultando que los expertos aceptaron la propuesta realizada por lo que queda avalado el procedimiento de trabajo para ser usado en el desarrollo de aplicaciones Web que utilicen el CMS Drupal.

Con el estudio realizado se da cumplimiento al objetivo propuesto creando una propuesta de un procedimiento de trabajo para el desarrollo de Portales Web utilizando el CMS Drupal.

RECOMENDACIONES

Se recomienda:

- Aplicar el procedimiento de trabajo en el desarrollo de Portales Web en proyectos de desarrollo que utilicen el CMS Drupal para que logren una mayor eficiencia y organización.
- Actualizar los artefactos definidos en el procedimiento de trabajo según se actualicen los artefactos definidos en el Expediente de Calidad de Proyectos de la Universidad.
- Generalizar el procedimiento definido de manera que pueda ser utilizado en la creación de Portales Web que utilicen otro CMS.

BIBLIOGRAFÍA

1. **Cuerda García, Xavier; Minguillón Alfonso, Julián.** *mosaic. mosaic.* [En línea] [Citado el: 13 de enero de 2009.] <http://mosaic.uoc.edu/articulos/cms1204.html>.
2. **De Troyer, Leune.** *WSDM: A User Centered Design Method for Web Sites.* 1997. Tilburg University, Infolab.
3. **Escalona.** *Metodologías de Desarrollo de Software.* 2001. [En línea] [Citado el: 14 de enero de 2009.] <http://www.lsi.us.es>.
4. **Escalona.** *NDT. Modelos y Técnicas para la Especificación y el Análisis de la Navegación en Sistemas Software.* 2004. Department of Computer Language and Systems. University of Seville.
5. **Escalona, Cuaresma, María José and González Romano, José Mariano.** [En línea] [Citado el: 16 de enero de 2009]. <http://www.lsi.us.es/docencia/get.php?id=2086>.
6. **Febles.** *Método Delphy.* 2003
7. **Fowler.** *Metodologías Ágiles: eXtreme Programming.* 2000.
8. **Jacobson, I.; Booch, G. y Rumbaugh, J.;** “El Proceso Unificado de Desarrollo de software”. 2000. Addison-Wesley.
9. **Koch.** *Software Engineering for Adaptative Hypermedia Applications.* UWE. 2001. Uni-Druck Publishing Company.
10. **Martin, Roberkys, Enriquez, Derick y Viltres, Hubert.** *Trabajo Investigativo sobre Sistemas de Gestión de Contenidos. CMS Drupal.* 2006.

11. **Pinton, Lic. Keyttia, y otros.** Propuesta Lineamientos Generales para el Desarrollo de Portales Web. 2006.
12. **Rational Software Corporation.** RUP. Rational Unified Process. 2003.
13. **Robertson.** Sistemas de Gestión de Contenidos. 2003. [En línea] [Citado el: 16 de enero de 2009.] <http://drupal.org>.
14. **WebML Group.** The Web Modeling Language. 2003. [En línea] [Citado el: 13 de enero de 2009.] <http://www.webml.org/webml/page3.do?ctx1=EN>.
15. Alcance de un Proyecto. [En línea] [Citado el 26 de febrero de 2009.] http://operations.blogs.ie.edu/archives/2007/06/la_gestion_del.php.
16. Análisis sobre Drupal. CMS Drupal. [En línea] [Citado el: 14 de enero de 2009.] http://www.webtaller.com/maletin/articulos/analisis_drupal.php.
17. Criterio de Expertos: Método Delphy. 2006
18. Estimación del Costo de un Proyecto. [En línea] [Citado el: 26 de febrero de 2009.] <http://iaap.wordpress.com/2008/02/18/la-estimacion-del-costo-de-un-proyecto>.
19. History. Drupal. [En línea] [12 de enero de 2009] <http://drupal.org/node/769>.
20. Navigational Development Techniques. [En línea] [Citado el: 23 de enero de 2009.] <http://lsi.ugr.es/~gedes/actividades/Dolmen4/a11.pdf>.
21. WebML. [En línea] [Citado el: 23 de enero de 2009.] <http://www.webml.org/webml/page1.do>.

GLOSARIO DE TÉRMINOS

- **CSS:** (*Cascading Style Sheets*). Lenguaje de programación que describe el estilo gráfico de un documento web.
- **GNU/GPL:** (*GNU General Public License*). Licencia de software libre ampliamente usada.
- **Hipertexto:** Vínculo en una o varias palabras de un texto o una imagen que remite a otra página web donde se puede encontrar información complementaria.
- **HTML:** (*HyperText Markup Language*). Lenguaje estándar para describir el contenido y la apariencia de las páginas en el WWW.
- **JSP:** (*Java Server Pages*). Tecnología que permite la creación de documentos HTML de forma dinámica en un servidor.
- **LDAP:** (*Lightweight Directory Access Protocol*). Protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.
- **Módulo:** Un módulo es una colección de funciones que se enlazan dentro de Drupal, ofreciendo funcionalidades adicionales.
- **PHP:** (*Hypertext Preprocessor*). Lenguaje de programación para el desarrollo de webs dinámicas, con sintaxis parecida a la C. Originalmente se conocía como Personal Home Page tools, herramientas para páginas personales (en Internet).
- **PSP:** Extensión de los ficheros creados por Adobe Photoshop.
- **RSS:** (*Really Simple Syndication*). Formato XML formulado para canalizar las noticias de un sitio web especializado en noticias.
- **UML:** Lenguaje Unificado de Modelado (*Unified Modeling Language*). Notación gráfica utilizada para describir sistemas de software.

- **WAI:** (*Web Accessibility Initiative*). Rama del *World Wide Web Consortium* que vela por la accesibilidad de la Web.
- **WYSISWG:** (*What You See Is What You Get*). Técnica que ofrece la reproducción exacta en pantalla de un texto tal como aparecería después en formato impreso.
- **XHML:** (*Extensible Hypertext Markup Language*). Reformulación del HTML 4.0 con XML 1.0. Toma la capacidad de formato del HTML y la une con la formalidad y extensibilidad del XML.
- **XML:** Lenguaje de Marcas Extensible (*eXtensible Markup Language*). Estándar de Internet que permite una estructura lógica y flexible para archivos de documentos.

ANEXOS

- **Anexo 1:** Estructura del Diagnóstico del Estudio de Sitios Web

Diagnóstico del Estudio de Sitios Web

<Nombre del proyecto>

<Nombre del producto>

<Versión>

Control de Versiones

Fecha	Versión	Descripción	Autor
<día/mes/año>	<x.x>	<detalles>	<nombre>

Reglas de Confidencialidad

Clasificación: <<Clasificación>>

Tabla de Contenidos

Introducción

Propósito

[Resumen del propósito de este documento]

Alcance

[Breve descripción del alcance del documento]

Definiciones, Acrónimos y Abreviaturas

Referencias

[Lista de documentos a los que se hace referencia]

Código	Título
[1]	Documento 1
[2]	Documento 2
[3]	Modelo de Diseño - Módulo de Administración v0.0

Glosario

[En el glosario aparecen un grupo de términos básicos]

Objetivos del Diagnóstico

Descripción de los Sitios Web a Revisar

[Se enuncian los objetivos que se persiguen con la realización del diagnóstico]

Resumen de diagnóstico por indicadores

[Resultado del diagnóstico en las entidades atendiendo a los indicadores seleccionados]

<Sitio Web 1>

<Indicador 1>

[Se pueden incluir entre los indicadores: los servicios e información que brindan los sitios Web, la estructura de información, el diseño y forma de presentar los contenidos, entre otros.]

Análisis de los Resultados

[Análisis de los resultados obtenido en el diagnóstico a los sitios Web. Se realiza un resumen de los aspectos más significativos a tener en cuenta]

Conclusiones

