

Universidad de las Ciencias Informáticas

FACULTAD 9



**IMPLEMENTACIÓN DE SERVICIOS PARA LA
GESTIÓN DE INFORMACIÓN EN EL POLO
PETROSOFT**

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN
CIENCIAS INFORMÁTICAS**

AUTOR(ES): Aymelis González Almora

Pedro Abigantús Pérez

TUTOR(A): Ing. Zaylí Rodríguez Luis

CIUDAD DE LA HABANA 2010, CUBA

AÑO DEL 52 ANIVERSARIO DEL TRIUNFO DE LA REVOLUCIÓN

“La inteligencia consiste no sólo en el conocimiento, sino también en la destreza de aplicar los conocimientos en la práctica.”

Aristóteles

Declaración de Autoría

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Facultad 9 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Aymelis González Almora

Firma del Autor

Pedro Abigantús Pérez

Firma del Autor

Ing. Zaylí Rodríguez Luis

Firma del Tutor

La información es un conjunto de datos que componen un mensaje sobre cierto fenómeno en determinado contexto. Permite resolver problemas y tomar decisiones, ya que su uso es la base del conocimiento. En la actualidad, los volúmenes de información existentes son elevados y la gestión de la misma llega, en ocasiones, a convertirse en un proceso tedioso si no se realiza de forma correcta.

Con el propósito de elevar el nivel de calidad en la gestión de información y en las tareas que se realizan en La Universidad de las Ciencias Informáticas, específicamente en el Polo Petrosoft de la Facultad 9, la presente investigación se centró en la implementación de los servicios de solicitud, diseminación selectiva y gestión de información del Polo Petrosoft.

En el presente trabajo se plasman los conceptos asociados al dominio del problema, se explica de forma detallada el análisis realizado para la selección de las herramientas que se utilizan en la implementación de los servicios de solicitud, diseminación y gestión de información y se muestra como se realizó la misma.

Además, se muestra el resultado arrojado en la encuesta realizada a los especialistas en el tema, los cuales eran los encargados de realizar la validación de los servicios implementados y de emitir un criterio final sobre los mismos.

Palabras Claves:

Diseminación Selectiva, Gestión de Información, Información, Servicios.

ÍNDICE DE TABLAS

Tabla 1: Diferencias entre metodologías ágiles y tradicionales..... - 17 -

Tabla 2: Comparación entre las Metodologías AUP, XP, RUP, MSF. - 21 -

Tabla 3: Definición de actores del negocio - 37 -

Tabla 4: Definición de trabajadores del negocio. - 39 -

Tabla 5: Descripción textual del caso de uso Atender Solicitud de Información..... - 41 -

Tabla 6: Descripción textual del caso de uso Recibir Información de Trabajos Científicos .. - 42 -

Tabla 7: Descripción textual del caso de uso Confeccionar Boletín. - 43 -

Tabla 8: Definición de Actores del Sistema..... - 47 -

Tabla 9: Descripción del Caso de Uso Solicitar Información Científica. - 49 -

Tabla 10: Descripción del caso de Uso Gestionar Solicitudes de Información Científica. - 49 -

Tabla 11: Descripción del Caso de Uso Gestionar Listado de Solicitudes de Información Científica. - 49 -

Tabla 12: Descripción del Caso de Uso Gestionar Información. - 50 -

Tabla 13: Descripción del Caso de Uso Subir Trabajo Científico a la Aplicación. - 50 -

Tabla 14: Descripción del Caso de Uso Gestionar Listado de Trabajos Científicos. - 50 -

Tabla 15: Descripción del Caso de Uso Gestionar Cronograma de Presentaciones..... - 51 -

Tabla 16: Descripción del Caso de Uso Gestionar Evento..... - 51 -

Tabla 17: Descripción del Caso de Uso Visualizar Cronograma de Presentaciones..... - 51 -

Tabla 18: Descripción del Caso de Uso Visualizar Publicación..... - 51 -

Tabla 19: Descripción del Caso de Uso Recibir Información Selectiva. - 52 -

Tabla 20: Descripción de la Tabla Solicitudes. - 64 -

Tabla 21: Descripción de la Tabla node_dir_listing_content. - 65 -

Tabla 22: Descripción de la Tabla Evento..... - 66 -

Tabla 23: Resultados de la encuesta realizada a los expertos..... - 71 -

ÍNDICE DE FIGURAS

Ilustración 1: Diagrama de caso de uso del negocio: Atender Solicitud de Información.....	- 38 -
Ilustración 2: Diagrama de caso de uso del negocio: Recibir Información de Trabajos Científicos.....	- 38 -
Ilustración 3: Diagrama de caso de uso del negocio: Confeccionar Boletín.	- 38 -
Ilustración 4: Modelo de objetos	- 39 -
Ilustración 5: Diagrama de Casos de Uso de Sistema.	- 48 -
Ilustración 6: Arquitectura del CMS Drupal	- 54 -
Ilustración 7: Diagrama de Paquetes del Sistema.....	- 55 -
Ilustración 8: Diagrama de clases del diseño de Drupal.....	- 57 -
Ilustración 9: Diagrama de clases del diseño del caso de uso Solicitar Información.	- 58 -
Ilustración 10: Diagrama de clases del diseño Gestionar Solicitudes de Información.	- 58 -
Ilustración 11: Diagrama de clases del diseño Gestionar Listado de Solicitudes de Información. -	59 -
Ilustración 12: Diagrama de clases del diseño del caso de uso Gestionar Información.....	- 59 -
Ilustración 13: Diagrama de Clases Persistentes.....	- 62 -
Ilustración 14: Modelo de Datos del Sistema.	- 63 -
Ilustración 15: Diagrama de Componentes del Caso de Uso Solicitar Información Científica.-	66 -
Ilustración 16: Diagrama de Componentes del Caso de Uso Gestionar Solicitudes de Información Científica.	- 67 -
Ilustración 17: Diagrama de componentes del caso de uso Gestionar Listado de Solicitudes de Información Científica.	- 67 -
Ilustración 18: Diagrama de componentes del caso de uso Gestionar Información.	- 67 -
Ilustración 19: Diagrama de Despliegue del Sistema.	- 68 -

ÍNDICE DE CONTENIDOS

INTRODUCCIÓN..... - 6 -

CAPÍTULO I: FUNDAMENTACIÓN TEÓRICA. - 11 -

1.1. INTRODUCCIÓN. - 11 -

1.2. CONCEPTOS FUNDAMENTALES. - 11 -

 1.2.1. Servicios..... - 11 -

 1.2.2. Gestión..... - 12 -

 1.2.3. Dato. - 13 -

 1.2.4. Información. - 13 -

 1.2.5. Gestión de la Información..... - 14 -

 1.2.6. Diseminación..... - 14 -

 1.2.7. Diseminación selectiva..... - 14 -

1.3. CONCLUSIONES..... - 15 -

CAPÍTULO II: TENDENCIAS Y TECNOLOGÍAS ACTUALES. - 16 -

2.1. INTRODUCCIÓN. - 16 -

2.2. Diferencias entre las metodologías ágiles y tradicionales..... - 16 -

2.3. Metodologías Tradicionales..... - 17 -

 2.3.1. Rational Unified Process. - 17 -

 2.3.2. Microsoft Solution Framework. - 18 -

2.4. Metodologías Ágiles. - 19 -

 2.4.1. Extreme Programming..... - 19 -

 2.4.2. RUP Ágil..... - 20 -

2.6. ¿Por qué AUP como metodología de desarrollo?..... - 21 -

2.7. Lenguaje de Modelado UML 2.0 y Herramienta CASE. - 22 -

 2.7.1. UML 2.0. - 22 -

 2.7.2. Visual Paradigm 6.4 For UML Enterprise Edition..... - 23 -

 2.7.3. Rational Rose Data Modeler..... - 23 -

2.8. ¿Por qué Visual Paradigm 6.4 For UML Enterprise Edition como herramienta CASE de modelado?..... - 24 -

2.9. Gestor de base de datos. - 24 -

 2.9.1. PostgreSQL 8.2.4.1..... - 25 -

 2.9.2. MySQL. - 26 -

2.10.	¿Por qué PostgreSQL 8.2.4.1 como gestor de base de datos?	- 26 -
2.11.	Lenguaje de Programación para servicios WEB.....	- 27 -
2.11.1.	PHP 5.2.4.....	- 27 -
2.11.2.	ASP.....	- 27 -
2.11.3.	JSP.	- 28 -
2.12.	¿Por qué PHP 5.2.4 como lenguaje de programación?	- 28 -
2.13.	Entorno de Desarrollo Integrado (IDE).....	- 28 -
2.13.1.	Eclipse.	- 29 -
2.13.2.	NetBeans.	- 29 -
2.14.	¿Por qué NetBeans como IDE?	- 29 -
2.15.	Servidor Web.....	- 30 -
2.15.1.	Apache 2.0.....	- 30 -
2.16.	¿Por qué Apache como Servidor Web?.....	- 31 -
2.17.	Sistema de Gestión de Contenidos (CMS).	- 31 -
2.17.1.	Joomla.	- 31 -
2.17.2.	Drupal 6.9.....	- 32 -
2.18.	¿Por qué Drupal 6.9 como CMS?.....	- 33 -
2.19.	CONCLUSIONES.....	- 34 -
CAPÍTULO III: PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.		- 35 -
3.1	INTRODUCCIÓN.	- 35 -
3.2	Modelamiento del Negocio.	- 35 -
3.3	Descripción del negocio.	- 35 -
3.4	Objeto de Automatización.....	- 36 -
3.5	Actores, Trabajadores y entidades del negocio.	- 37 -
3.7.	Descripción textual de los casos de uso del negocio.	- 40 -
3.8.	Reglas del Negocio.	- 43 -
3.9.	Levantamiento de Requisitos.	- 44 -
3.10.	Patrones de Casos de Uso.....	- 46 -
3.11.	Descripción del Sistema Propuesto.	- 47 -
3.12.	Diagrama de casos de uso del sistema.	- 48 -
3.14.	CONCLUSIONES.....	- 52 -
CAPÍTULO IV: IMPLEMENTACIÓN DE LA SOLUCIÓN PROPUESTA.		- 53 -

4.1	INTRODUCCIÓN.	- 53 -
4.2	Arquitectura.....	- 53 -
4.3	Diseño.....	- 55 -
4.3.1	Diagramas de Paquetes.....	- 55 -
4.3.2	Diagrama de clases del diseño.....	- 56 -
4.3.3	Patrones de Diseño.....	- 60 -
4.3.4	Diseño de la Base de Datos.	- 62 -
4.3.5	Diagrama de componentes del sistema.....	- 66 -
4.3.6	Diagrama de Despliegue del Sistema.....	- 68 -
4.4	CONCLUSIONES.	- 68 -
CAPÍTULO V: VALIDACIÓN DE LA PROPUESTA.....		- 69 -
5.1	INTRODUCCIÓN.	- 69 -
5.2	Análisis de la validación.	- 69 -
5.3.	Elaboración de la encuesta.	- 70 -
5.4.	Procesando los datos.	- 71 -
5.6	CONCLUSIONES.....	- 72 -
CONCLUSIONES GENERALES.		- 73 -
RECOMENDACIONES.....		- 74 -
TRABAJOS CITADOS.....		- 75 -
BIBLIOGRAFÍA.....		- 77 -
GLOSARIO DE TÉRMINOS.....		- 78 -

INTRODUCCIÓN.

Desde tiempos inmemoriales, tener la información justa en el momento oportuno ha significado “poder” y solamente las personas o grupos de personas que se podían permitir financiar ese intercambio de información disfrutaron de ese privilegio. Sin embargo, con el nacimiento de Internet ha disminuido drásticamente el tiempo que transcurre entre la necesidad y su ‘realización’, permitiendo alcanzar nuevos niveles de capacidad individual.(Saint-Paul 2008)

La información es un recurso valioso que debe estar accesible para todos los usuarios. En el mundo actual donde los cambios son acelerados y continuos, se ha producido una explosión de la información en todos los sectores sociales, por tanto, es preciso realizar el enfoque de estos servicios cada vez más a los clientes, logrando que la información que reciba cada uno sea verdaderamente útil y relevante.

Para las empresas la información es un recurso esencial, y su gestión ocupa un espacio mayor en la economía de los países a escala mundial; existiendo una estrecha relación entre la gestión de la información y la calidad del trabajo en una organización. Este motivo hace que sea importante una adecuada gestión, conservación y difusión de la misma.

Existen en el mundo muchas organizaciones y entidades que se preocupan por realizar una correcta gestión de la información y promueven además, la cultura de optimización de recursos en las empresas.

Cuba es uno de los países que cuenta con entidades que se dedican a gestionar información. Un ejemplo de esto es la empresa CUBAENERGÍA (Centro de Gestión de la Información y Desarrollo de la Energía): entidad encargada de gestionar información relacionada con la energía. También se cuenta con otros centros como IDICT (Instituto de Información Científica y Tecnológica): empresa líder de la industria de la información en Cuba, que ofrece soluciones de gestión de información y del conocimiento a organizaciones y comunidades.

Cuba no solo cuenta con entidades dedicadas a ofrecer estos servicios, también existen cátedras universitarias que forman parte de este grupo y que colaboran con otros países. Un ejemplo de esto es la cátedra UNESCO, de Gestión de la Información en las Organizaciones en el marco del Grupo Santander, establecida en 1993 en la Universidad de la Habana.

Aunque se cuenta con entidades que conocen del tema existen aún en la actualidad otras muchas que no tienen conocimiento de lo que es la gestión de la información o que presentan deficiencias en este proceso.

La Universidad de las Ciencias Informáticas (UCI) es una universidad productiva, cuya misión es desarrollar software y servicios informáticos a partir de la vinculación estudio – trabajo que propone el modelo de formación centrado en el aprendizaje. La producción de software y servicios informáticos se basa en la integración de los procesos de formación, investigación y producción en torno a una temática para convertirla en una rama productiva. Este espacio de integración temática es denominado Polo Productivo. Aquí se promueve la formación de pregrado, postgrado, la colaboración nacional e internacional, el fomento de líneas de investigación y desarrollo y la ejecución de proyectos en el marco de acuerdos de trabajo. La producción se concentra en el desarrollo de proyectos en más de 10 facultades que cuentan con 30 Polos Productivos y se destacan resultados en las esferas nacionales de salud, educación, software libre, teleformación, sistemas legales, realidad virtual, automatización, bioinformática, procesamiento de imágenes y señales, entre otras. El desarrollo de proyectos, genera gran cantidad de documentación. Cuando la documentación es mucha, es recomendable realizar un correcto proceso de gestión de información para obtener una mejor organización y calidad en las tareas.

El Polo Petrosoft de la Facultad 9 es uno de los que se encuentra vinculado a la producción. En el Polo se genera gran cantidad de documentos. Los mismos no son bien gestionados o se gestionan de forma manual, por lo que se torna un proceso tedioso el trabajo a realizar; debido al gran volumen existente en la actualidad. Trayendo consigo atraso en los servicios que se prestan relacionados con este proceso.

Entre los servicios que se encuentran más afectados está el de solicitud de información de los Jefes de Polo al Grupo de Gestión de la Información, integrado por estudiantes y profesores. En la actualidad el proceso de solicitud de información se realiza manualmente, siendo los estudiantes los que recopilan esta información para después responder a las solicitudes. También existen problemas con la actualización frecuente de la información relacionada con las solicitudes, pues cuando algún Jefe de Grupo del Polo necesita información y quiere recibir frecuentemente actualizaciones de la misma hay que llevar estadística sobre esto y este proceso se realiza de forma manual.

Otra dificultad es cuando se realizan eventos científicos en el Polo, ya que existen problemas al subir la información a la dirección que se indica. Actualmente todo el flujo de información se realiza a través del correo electrónico por lo que se puede dar el caso que exista pérdida de la misma o la información no llegue al destinatario correcto.

El Polo Petrosoft además, presenta problemas en su intranet, ya que no permite leer el contenido de otras páginas sin necesidad de un navegador, o lo que es lo mismo no utiliza un software diseñado para leer estos contenidos. En la actualidad el uso de canales de sindicación (canales RSS) es muy útil pues permite mantener actualizado los sitios web e incluso los usuarios, este motivo hace que sea recomendable el uso de los mismos.

Teniendo en cuenta la problemática anteriormente explicada se plantea el siguiente **Problema a Resolver**: Ineficiencia en los servicios de solicitud, disseminación selectiva y gestión de información existentes en el Polo Productivo Petrosoft de la facultad 9.

Para dar solución al problema planteado se propone como **Objeto de Estudio**: El proceso de gestión de la información.

Este se enmarca en el siguiente **Campo de Acción**: La informatización de los procesos de solicitud, disseminación selectiva y gestión de información.

Planteándose como **Objetivo General** para resolver el problema: Implementar los servicios de solicitud, disseminación selectiva y gestión de información del Polo Petrosoft para mejorar el desarrollo de las tareas asociadas a estos procesos.

Para dar cumplimiento al objetivo general se trazan las siguientes **Tareas de la Investigación**:

1. Describir el estado actual de los procesos de gestión de la información.
2. Describir el estado actual de las principales tecnologías de desarrollo para implementar los servicios de gestión de la información.
3. Implementar los servicios de solicitud, disseminación selectiva y gestión de información para el Polo Petrosoft.
4. Evaluar los resultados.

La **Idea a defender** que se establece en la presente investigación es la siguiente:

Si se implementan los servicios de solicitud, disseminación selectiva y gestión de información se logrará mejorar las tareas que se realizan en el Polo Petrosoft relacionadas con estos servicios.

Los **métodos científicos** que se utilizan en el desarrollo de la investigación son los teóricos y los empíricos.

Dentro de los métodos teóricos que se utilizan se encuentran:

- Analítico-Sintético: Se utiliza en el estudio y comprensión de toda la información recopiladas para llegar a conclusiones válidas y necesarias para el desarrollo de la investigación
- Análisis Histórico Lógico: Para estudiar el estado del arte y la evolución del proceso de gestión de información.

Dentro de los métodos empíricos que se utilizan se encuentran:

- Observación: Para observar cómo se realiza el proceso de solicitud y gestión de la información actualmente en el Polo Petrosoft de la facultad 9 de la Universidad de las Ciencias Informáticas.
- Entrevista: Para entrevistar al Jefe de Grupo de Desarrollo Humano y determinar cuáles son los servicios que actualmente presentan deficiencias y necesitan ser automatizados, además, el resultado arrojado por estas se utilizará en la confección de los formularios de solicitud y diseminación selectiva de información.

Además, se empleó dicho método para que las personas expertas realizaran la validación de los servicios. Los resultados arrojados en la encuesta de validación se emplearon para determinar si los servicios desarrollados estaban aptos para ser utilizados en el Polo.

La implementación de los servicios de solicitud, diseminación selectiva y gestión de la información será realizada con las herramientas seleccionadas a partir de un estudio detallado efectuado a las mismas. Esto logrará un mejor desarrollo en los procesos descritos anteriormente ya que no se realizarán de forma manual y su desarrollo será de manera rápida y eficiente.

El presente trabajo estará estructurado de la siguiente manera:

CAPÍTULO I: FUNDAMENTACIÓN TEÓRICA.

Este capítulo incluye una fundamentación teórica de todos los elementos y conceptos relacionados con el problema a resolver para un mejor entendimiento del mismo.

CAPÍTULO II: TENDENCIAS Y TECNOLOGÍAS ACTUALES.

Este capítulo contiene la comparación entre las diferentes metodologías, herramientas y lenguajes de desarrollo actual que pueden ser utilizados para darle solución al problema. Además, se expone la selección de las herramientas a utilizar y el porqué de esto.

CAPÍTULO III: MODELAMIENTO DEL NEGOCIO.

Este capítulo incluye la descripción detallada del negocio. Contiene los diagramas de actividades y de objetos. Contiene el glosario de término para un mejor entendimiento de las partes que interactúan en el negocio, se determinan las reglas del negocio y los requisitos funcionales y no funcionales del sistema.

CAPÍTULO IV: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

Este capítulo incluye el modelo de diseño del sistema, la primera versión del prototipo interfaz de usuario, los diagramas de clases del diseño del sistema y los diagramas de componentes. Contiene además el diseño de la base de datos, así como con las relaciones entre tablas, el modelo de datos, así como el modelo de despliegue del sistema.

CAPITULO V: VALIDACIÓN DE LA PROPUESTA.

Este capítulo incluye el resultado arrojado por la encuesta realizada a los expertos que validaron el sistema y los aspectos que se tuvo en cuenta para la selección de los mismos. Se muestra además los resultados parciales y generales tras el procesamiento de los datos.

CAPÍTULO I: FUNDAMENTACIÓN TEÓRICA.

1.1. INTRODUCCIÓN.

Este capítulo tiene como objetivo describir teóricamente los principales conceptos relacionados con el problema a resolver y que van a servir como base para un mejor entendimiento y desarrollo del presente trabajo.

1.2. CONCEPTOS FUNDAMENTALES.

1.2.1. Servicios.

Es la relación que se establece entre dos partes, en la que una (proveedor) organizada sistémicamente, ofrece a la otra (usuario) los resultados de la también relación que se produce entre los recursos con que cuenta este proveedor, tanto financieros como físicos o tecnológicos pero, fundamentalmente, los de información y los humanos, estos últimos, adiestrados previamente en la gestión de todos los anteriores. En todas estas relaciones median procesos de comunicación e informacionales en general y se llevan a cabo con el objetivo de satisfacer las necesidades de información tanto implícitas como explícitas de la parte receptora cuyo conocimiento por el proveedor también está implícito en estas relaciones las que pudiendo ser lucrativas o no, en última instancia están encaminadas a, con sus resultados, elevar el nivel de aptitud de la parte usuaria, individual o colectiva, para abordar problemas tanto personales como sociales.(Dante 2004)

Kotler, Bloom y Hayes, definen un servicio de la siguiente manera: "Un servicio es una obra, una realización o un acto que es esencialmente intangible y no resulta necesariamente en la propiedad de algo. Su creación puede o no estar relacionada con un producto físico."(Kotler Philip 2004)

Stanton, Etzel y Walker, definen los servicios "como actividades identificables e intangibles que son el objeto principal de una transacción ideada para brindar a los clientes satisfacción de deseos o necesidades."(Stanton William 2004)

Para Richard L. Sandhusen, "los servicios son actividades, beneficios o satisfacciones que se ofrecen en renta o a la venta, y que son esencialmente intangibles y no dan como resultado la propiedad de algo."(Sandhusen 2002)

Según Lamb, Hair y McDaniel, "un servicio es el resultado de la aplicación de esfuerzos humanos o mecánicos a personas u objetos. Los servicios se refieren a un hecho, un desempeño o un esfuerzo que no es posible poseer físicamente".(Charles Lamb 2002)

Para la American Marketing Association (A.M.A.), los servicios (según una de las dos definiciones que proporcionan) son "productos, tales como un préstamo de banco o la seguridad de un domicilio, que son intangibles o por lo menos substancialmente. Si son totalmente intangibles, se intercambian directamente del productor al usuario, no pueden ser transportados o almacenados, y son casi inmediatamente perecederos. Los productos de servicio son a menudo difíciles de identificar, porque vienen en existencia en el mismo tiempo que se compran y que se consumen. Abarcan los elementos intangibles que son inseparabilidad; que implican generalmente la participación del cliente en una cierta manera importante; no pueden ser vendidos en el sentido de la transferencia de la propiedad; y no tienen ningún título. Hoy, sin embargo, la mayoría de los productos son en parte tangibles y en parte intangibles, y la forma dominante se utiliza para clasificarlos como mercancías o servicios (todos son productos). Estas formas comunes, híbridas, pueden o no tener las cualidades dadas para los servicios totalmente intangibles".(Thompson 2006)

Un servicio no es más que la actividad intangible que realizan dos o más personas donde una parte satisface las necesidades o los deseos de la otra.

1.2.2. Gestión.

La Real Academia Española define gestión como acción y efecto de gestionar. (Española 2009)
Según Gustavo Rebolledo gestión no es más que la capacidad de la institución para definir, alcanzar y evaluar sus propósitos, con el adecuado uso de los recursos disponibles.(Saavedra 2009)

Pablo Murray expresa que la gestión no es más que la herramienta que engloba fases de la administración siendo la gestión misma un elemento con desarrollo de la administración. (Murray 2002)

Según Roberto Faga gestión es interactuar en todas las áreas de una empresa, organización, unidades, instituciones informativas .Actividad dirigida a obtener y asignar los recursos necesarios para el cumplimiento de los objetivos de una organización. (Murray 2002)

Gestión no es más que la actividad realizada por determinada institución para obtener sus propósitos con el uso adecuado de los recursos que posee.

1.2.3. Dato.

Se define como dato los hechos y cifras que se pueden utilizar como base para el razonamiento, la discusión o el cálculo. (Romero Heber 2009)

Sergio D' Ambrosio esclarece que “por sí mismos los datos no tienen capacidad de comunicar un significado y por tanto no pueden afectar el comportamiento de quien los recibe. Para ser útiles, los datos deben convertirse en información para ofrecer un significado, conocimiento, ideas o conclusiones.”(Romero Heber 2009)

En sus investigaciones, Sanguino conceptualiza la palabra dato como “elemento o cantidad conocida que sirve de base para la resolución de un problema. Principio en el que se asienta una discusión. Elemento o base para la formulación de un juicio. Lo que se presenta a la conciencia como inmediata, no construida o no elaborada. Elemento de información o representación de hechos o de instrucciones, en forma apropiada para el almacenamiento, procesamiento o transmisión por medios automáticos”. (Romero Heber 2009)

Según Lauro Soto un dato puede definirse como la unidad mínima de información o bit, puede ser un caracter, una palabra, etc. (Romero Heber 2009)

Los datos no tienen significado sin tener una referencia de espacio y tiempo. Es algo fuera de contexto, que no tiene relación con otra cosa. Generalmente se les asigna significado al asociarlos a algo.

1.2.4. Información.

En sus investigaciones Sanguino describe el término información como “acto o efecto de informar. Datos acerca de alguien o de algo. Comunicación o noticia traída al conocimiento de una persona o del público. Instrucción, dirección. Según la teoría de la información, medida de reducción de la incerteza sobre un determinado estado de cosas, por intermedio de un mensaje.”(Romero Heber 2009)

En un estudio realizado por Pablo Murray este dice que la información es la agrupación de datos organizados y presentados en un contexto. (Murray 2002)

Para Sergio D' Ambrosio la información “es un conjunto de datos significativos y pertinentes que describen sucesos o entidades.”(Romero Heber 2009)

El término información, según Verónica Sorberamurina, “alude a un conjunto de datos organizados de manera tal que portan o arrojan un significado.”(Romero Heber 2009)

Según Gloria Ponjuán Dante información es: “Mensaje, generalmente en forma de un documento o de una comunicación audible o visible con un emisor y un receptor”. La

información está compuesta por datos los cuales reafirman la misma o le aportan valor. (Dante 2004)

La información no es más que datos asociados a un contexto determinado los cuales van a suministrar o arrojar algún significado.

1.2.5. Gestión de la Información.

Phil Bartle plantea que la gestión de la información es el proceso de analizar y utilizar la información que se ha recabado y registrado para permitir a los administradores (de todos los niveles) tomar decisiones documentadas.(Bartle 2009)

Gloria Ponjuán Dante define gestión de la información al proceso mediante el cual se obtienen, despliegan o utilizan recursos básicos (económico, físicos, humanos, materiales) para manejar información dentro y para la sociedad a la que sirve. (Dante 2004)

Gilberto Sotolongo expresa que "la gestión de la información se ocupa de los resultados finales, no sólo de citas y localizaciones". Fernández-Molina lo corrobora cuando afirma que los profesionales de la información y sus instituciones son un factor indispensable para la permanencia de la organización en el mercado. (Dante 2004)

Carlota Bustelo y Raquel Amarilla coinciden al definir la gestión de la información como el conjunto de actividades realizadas con el fin de controlar, almacenar y, posteriormente, recuperar adecuadamente la información producida, recibida o retenida por cualquier organización en el desarrollo de sus actividades. (Bustelo 2001)

La gestión de información comprende las actividades relacionadas con la obtención de la información adecuada, a un precio adecuado, en el tiempo y lugar adecuado, para tomar la decisión adecuada. (Woodman 1985)

La gestión de la información no es más que el proceso que se realiza en las organizaciones a fin de llevar el control de todas las actividades efectuadas a la información.

1.2.6. Diseminación.

La Real Academia Española define diseminación como acción o efecto de diseminar.(Española 2009)

1.2.7. Diseminación selectiva.

William James Struder plantea que la diseminación selectiva es la diseminación de nueva información a individuos o grupos de acuerdo a sus intereses establecidos.(Vallado 2008)

Hans Peter Luhn lo define como el servicio(dentro de una organización) dedicado a canalizar nuevos elementos de información(a partir de cualquier fuente) a aquellos puntos dentro de la misma organización donde la probabilidad de su utilidad en el trabajo diario o de su interés sea elevada.(Vallado 2008)

Por su parte, Miguel Francisco Ibáñez González, Doris Beatriz Rivera Ibarra, Ignacio Devesa Gutiérrez y Martha Ibáñez Marmolejo señalan que el sistema de diseminación selectiva se puede definir como: “un servicio especializado dentro del quehacer bibliotecario, que se encarga de seleccionar las informaciones nuevas, cualquiera que sea la fuente y ser canalizadas a los integrantes de una organización donde se encuentre dicho sistema bibliotecario y que tendrá grandes probabilidades de ser útil en la capacitación del personal, perfeccionamiento y aumento de la productividad. Este medio permite que la divulgación de la información sea de manera selecta, prioritaria y cubra las necesidades específicas del usuario al que va dirigida” .(Vallado 2008)

Judith Holt Conner lo describe como cualquier procedimiento manual o automatizado que proporciona un servicio personalizado de alerta, seleccionando la nueva literatura de probable relevancia para cada individuo, de acuerdo a sus necesidades de investigación. (Vallado 2008)

La diseminación selectiva no es más que el proceso de seleccionar, según las necesidades del usuario, la información de interés para este y proporcionársela sistemáticamente.

1.3. CONCLUSIONES.

La correcta gestión de la información ofrece un entorno de trabajo más organizado.

Permite tener un mayor control de todas las actividades realizadas en la organización a la información a fin de disminuir el tiempo de realización de muchas tareas.

Su principal beneficio está en permitir el acceso a toda la información que se genera en cada una de las unidades de la organización, el control de versiones, así como la homogenización de esta información, trayendo consigo un ahorro considerable de los recursos utilizados para el almacenamiento de los archivos y una mejora de los procesos organizacionales.

La diseminación selectiva por otra parte, permite mantener actualizada a las personas según sus intereses, evitando malgasto de recursos o abrumando a las mismas con informaciones que no sean del interés de estas.

Estos conceptos bien aplicados en una organización permiten mejorar su funcionamiento y aumentan la calidad del trabajo en la misma.

CAPÍTULO II: TENDENCIAS Y TECNOLOGÍAS ACTUALES.

2.1. INTRODUCCIÓN.

En el desarrollo de software, la selección adecuada de la metodología de desarrollo es uno de los aspectos fundamentales para construir un producto con la calidad requerida y para satisfacer las necesidades de los clientes. Las metodologías facilitan el trabajo y proveen una trayectoria a seguir para la correcta evolución y desarrollo de los sistemas; lo hacen desarrollando un proceso detallado inspirado por otras disciplinas de la Ingeniería de Software. Las metodologías han existido desde hace mucho tiempo atrás, pero su uso no había sido muy bien aceptado debido a la poca popularidad que presentaban. Con el transcurso del tiempo estas han sido adaptadas y modificadas de acuerdo a las necesidades de los desarrolladores, alcanzando a ser lo que son en el presente; facilitando y brindando un mejor desarrollo del software.

Las metodologías se pueden clasificar en ágiles y tradicionales. Presentando cada una características diferentes.

2.2. Diferencias entre las metodologías ágiles y tradicionales.

Las metodologías tradicionales, o como comúnmente se les denomina robustas, se centran especialmente en el control del proceso, mediante una exhaustiva documentación; definiendo roles, actividades, artefactos, herramientas y notaciones para el modelado y una documentación detallada. Estas metodologías son muy efectivas y necesarias en proyectos grandes.

Las metodologías ágiles en cambio, dan mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas. Con cortos documentos centrados en lo esencial. (Amaro Calderón 2007)

A continuación se presenta una tabla que ilustra las diferencias entre las metodologías tradicionales y ágiles.

Metodologías Tradicionales	Metodologías Ágiles
Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo.	Basadas en heurísticas provenientes de prácticas de producción de código.

Resistencia a los cambios.	Preparadas para cambios durante el proyecto.
Impuestas externamente.	Impuestas internamente (por el equipo).
Proceso más controlado, con numerosas políticas/normas.	Proceso menos controlado, con pocos principios.
Existe un contrato prefijado.	No existe contrato tradicional (bastante flexible).
El cliente se comunica con el equipo de desarrollo mediante reuniones.	El cliente es parte del equipo de desarrollo.
Grupos grandes y distribuidos.	Grupos pequeños (menos de 10 integrantes) trabajando en el mismo sitio.
Más artefactos.	Pocos artefactos.
Más roles.	Pocos roles.
La arquitectura de software es esencial y se expresa mediante modelos.	Menos énfasis en la arquitectura del software.

Tabla 1: Diferencias entre metodologías ágiles y tradicionales.

2.3. Metodologías Tradicionales.

Entre las metodologías tradicionales que existen se pueden encontrar: RUP (del inglés Rational Unified Process), MSF (del inglés Microsoft Solution Framework), entre otras.

2.3.1. Rational Unified Process.

La metodología RUP es una metodología creada por Jacobson, Rumbaugh y Booch. Unifica los mejores elementos de metodologías anteriores y está preparada para desarrollar proyectos grandes y complejos. Es orientada a objetos y utiliza UML como lenguaje de representación visual. El ciclo de vida de esta metodología se caracteriza por ser dirigido por casos de usos, centrado en la arquitectura e iterativo e incremental.

RUP divide en 4 fases el desarrollo del software:

- **Inicio:** El objetivo en esta etapa es determinar la visión del proyecto.
- **Elaboración:** En esta etapa el objetivo es determinar la arquitectura óptima.
- **Construcción:** En esta etapa el objetivo es llevar a obtener la capacidad operacional inicial.
- **Transmisión:** El objetivo es llegar a obtener el release del proyecto.

2.3.2. Microsoft Solution Framework.

Esta es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas. Se compone de varios modelos encargados de planificar las diferentes partes implicadas en el desarrollo de un proyecto: Modelo de Arquitectura del Proyecto, Modelo de Equipo, Modelo de Proceso, Modelo de Gestión del Riesgo, Modelo de Diseño de Proceso y finalmente el modelo de Aplicación (Sánchez 2004)

Se caracteriza por ser una metodología adaptable y de uso limitado a un específico lugar. Puede organizar equipos pequeños así como también equipos que requieren gran cantidad de personas. Es utilizada en el ambiente de desarrollo de cualquier cliente y para desarrollar soluciones basadas sobre cualquier tecnología.

El modelo de proceso MSF consta de 5 fases distintas:

- **Ideación:** Se reúne un equipo y define la visión y el ámbito de una solución que cumplirá los objetivos del cliente.
- **Planificación:** Se despliegan los procesos de diseño conceptual, lógico y físico, así como la especificación funcional.
- **Desarrollo:** Se realiza la mayor parte de la construcción de los elementos.
- **Estabilización:** El equipo realiza pruebas a la solución para resolver posibles errores y se preparan para el lanzamiento de producción.
- **Implementación:** Durante esta fase el equipo implanta la tecnología base y los componentes relacionados, estabiliza la instalación, traspasa el proyecto al personal soporte y operaciones, y obtiene la aprobación final del cliente.

2.4. Metodologías Ágiles.

Las metodologías ágiles son las más usadas en la actualidad debido a las numerosas ventajas que estas ofrecen. Existen muchos tipos de metodologías ágiles, ejemplo de estas son XP (del inglés EXTREME PROGRAMMING), AUP (del español RUP Ágil), entre otras.

2.4.1. Extreme Programming.

XP es una de las metodologías de desarrollo de software más triunfante en la actualidad. Es empleada en proyectos de corta duración y de pocos integrantes. En esta metodología el usuario final es parte del equipo de desarrollo y es flexible a los cambios que se producen durante el desarrollo del proyecto.

XP se caracteriza por basarse fundamentalmente en el desarrollo iterativo mostrando pequeñas mejoras consecutivamente. Esto se logra gracias a las pruebas que se realizan continuamente al producto, con el fin de prever posibles errores que pudiesen aparecer en el desarrollo del mismo. Brinda además la posibilidad de reutilizar el código, creando patrones o estándares que son más flexibles al cambio y promueve la responsabilidad de todos los integrantes del equipo de desarrollo para corregir y extender cualquier parte del proyecto, no limitando esto a una persona en específico.

El ciclo de vida ideal de XP consiste de seis fases:

Exploración: En esta fase, los clientes plantean los intereses más importantes y que son de utilidad para la primera entrega del producto. Al mismo tiempo, el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo.

Planificación de la Entrega (Release): En esta fase el cliente establece la prioridad de cada tarea. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente.

Iteraciones: Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado. En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada durante el resto del proyecto. Todo el trabajo de la iteración es expresado en tareas de programación, cada una de ellas es asignada a un programador como responsable, pero llevadas a cabo por parejas de programadores.

Producción: Esta fase requiere de pruebas adicionales y revisiones de rendimiento antes que el sistema sea trasladado al contexto del cliente. Se deben tomar decisiones sobre la adición de nuevas características a la versión actual, debido a posibles cambios durante esta fase.

Mantenimiento: Mientras la primera versión se encuentra en producción, el proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones.

Muerte del Proyecto: En esta fase es necesario que se satisfagan las necesidades del cliente en aspectos como son rendimiento y confiabilidad del sistema. Se genera la documentación final del sistema y no se realizan más cambios en la arquitectura. Esto también ocurre cuando no se generan los beneficios esperados por el cliente o cuando no hay presupuesto para mantener el sistema.

2.4.2. RUP Ágil.

AUP es una versión simplificada de RUP que utiliza técnicas y conceptos de este. Se basa en la gestión de riesgos proponiendo que aquellos componentes con alto riesgo tengan más prioridad que los demás y sean desarrollados en etapas tempranas del proyecto. Desarrolla prototipos ejecutables durante la fase de elaboración del producto, demostrando la validez de la arquitectura para los requisitos clave del producto y determinando los riesgos técnicos. Al igual que en RUP, en AUP se establecen cuatro fases que transcurren de manera consecutiva y que acaban con hitos claros alcanzados. El proceso AUP establece un modelo más simple que el que aparece en RUP por lo que reúne en una única disciplina las disciplinas de Modelado de Negocio, Requisitos y Análisis y Diseño. El resto de disciplinas (Implementación, Pruebas, Despliegue, Gestión de Configuración, Gestión y Entorno) coinciden con las restantes de RUP. (Fowler 2003)

RUP Ágil es flexible y propone los mismos roles y artefactos que RUP, solo que no hay necesidad de generar toda la documentación que se requiere en cada flujo de trabajo.

El ciclo de vida de AUP es el siguiente:

- **Modelo:** En esta fase el objetivo fundamental es entender el negocio de la organización, el problema de dominio que se aborda en el proyecto, y determinar una solución viable para resolver el problema de dominio.
- **Implementación:** Transformar el(los) modelo(s) en código ejecutable y realizar un nivel básico de pruebas individuales.

- **Prueba:** Realizar una evaluación objetiva para garantizar la calidad. Esto incluye la búsqueda de defectos, validar que el sistema funciona tal como está establecido, y verificar que se cumplan los requisitos.
- **Despliegue:** Realizar un plan para la presentación del sistema y ejecutarlo para hacer que el sistema se encuentre a disposición de los usuarios finales.
- **Gestión de Configuración:** Realizar la gestión de acceso a artefactos de su proyecto. Esto incluye no sólo el seguimiento de las versiones del artefacto en el tiempo, sino también el control y la gestión de cambios para ellos.
- **Gestión del Proyecto:** Dirigir las actividades que se lleva a cabo en el proyecto. Esto incluye la gestión de los riesgos, la dirección de personas (la asignación de tareas, el seguimiento de los progresos, entre otros), y coordinar con las personas para garantizar que se entrega a tiempo y dentro del presupuesto.
- **Ambiente:** Apoyar el resto de los esfuerzos por garantizar que el proceso adecuado, la orientación (normas y directrices), y herramientas (hardware, software, entre otros) están disponibles para el equipo según cuando ellos lo necesiten. (Velezmoro 2008)

2.5. Comparación entre las Metodologías AUP, XP, RUP, MSF.

Metodologías	Dirigido por casos de uso	Desarrollo iterativo e incremental	Participación activa del usuario	Adaptación del proceso	Centrado en la arquitectura
AUP	SÍ	SÍ	NO	SÍ	SÍ
XP	NO	NO	SÍ	SÍ	NO
RUP	SÍ	SÍ	NO	SÍ	SÍ
MSF	NO	NO	NO	SÍ	SÍ

Tabla 2: Comparación entre las Metodologías AUP, XP, RUP, MSF.

2.6. ¿Por qué AUP como metodología de desarrollo?

AUP es una metodología flexible que no requiere de una gran cantidad de desarrolladores. Es concisa en el aspecto de la documentación, permitiendo generar solo la necesaria y no la especificada para cada flujo de trabajo como lo hace RUP. Está diseñada para trabajar en

proyectos pequeños donde la atención se centra en las actividades que realmente son importantes. Permite el uso de herramientas de cualquier tipo, incluyendo aquí las de código abierto. Es fácil de manejar a través de herramientas de edición HTML sin necesidad de ser adaptada y es una metodología que se ajusta y aprovecha las ventajas que brindan las metodologías ágiles.

Aunque XP es una metodología popular y que brinda muchas ventajas presenta también desventajas. Ejemplo de estas es que no se genera documentación. En la actualidad en cualquier sistema por más simple que sea el cliente solicita que le sea entregado además del software manuales para los usuarios y otras documentaciones.

XP por otra parte, no es viable en el desarrollo de sistemas que no requieren de la presencia del usuario, pues este requiere que el usuario esté siempre presente como un miembro más del equipo de desarrollo. Además, al ser uno de los principios la programación en pareja, en esta metodología surge la interrogante de que si permite que los desarrolladores se encuentren en diferentes situaciones geográficas. Siendo esto un inconveniente para los desarrolladores.

De acuerdo con las características que presentan los servicios a desarrollar y tras el estudio realizado a las diferentes metodologías de desarrollo se determina como metodología de desarrollo a utilizar AUP.

2.7. Lenguaje de Modelado UML 2.0 y Herramienta CASE.

2.7.1. UML 2.0.

UML (del inglés Unified Modeling Language) es un lenguaje para visualizar, especificar, construir y documentar los elementos que componen un sistema desarrollado con la tecnología orientada a objetos. Se ha convertido en el estándar internacional para definir, organizar y visualizar los elementos que configuran la arquitectura de una aplicación orientada a objetos. Este lenguaje pretende unificar las experiencias acumuladas sobre técnicas de modelado e incorporar las mejores prácticas en un acercamiento estándar.

UML permite la creación de los diferentes modelos que ofrecen las vistas necesarias para la construcción de un software de calidad y permite la comprensión del sistema que se quiere realizar tanto por parte de los usuarios finales, como de los desarrolladores que implementarán la solución. (Rodríguez Luis 2009)

2.7.2. Visual Paradigm 6.4 For UML Enterprise Edition.

Es una herramienta CASE (Ingeniería de Software Asistida por Ordenador) que utiliza UML como lenguaje de modelado. Permite crear diferentes tipos de diagramas en un ambiente totalmente visual. Es muy sencillo de usar, fácil de instalar y actualizar. Genera código para varios lenguajes. Tiene integrado el MS Visio y es compatible con otras ediciones, posibilita un entorno de creación de diagramas para UML 2.x. (Paradigm 2007)

Visual Paradigm además se integra con herramientas Java como son Eclipse/IBM WebSphere, JBuilder, NetBeans IDE, Oracle JDeveloper, BEA Weblogic. (Hernandis 2005)

Características de Visual Paradigm 6.4 For UML Enterprise Edition.

- Soporte de UML versión 2.1.
- Diagramas de Procesos de Negocio - Proceso, Decisión, Actor de negocio, Documento.
- Ingeniería inversa - Código a modelo, código a diagrama.
- Ingeniería inversa Java, C++.
- Generación de código - Modelo a código, diagrama a código.
- Editor de Detalles de Casos de Uso - Entorno todo-en-uno para la especificación de los detalles de los casos de uso, incluyendo la especificación del modelo general y de las descripciones de los casos de uso.
- Generación de objetos Java desde la base de datos.
- Generación de bases de datos - Transformación de diagramas de Entidad-Relación en tablas de base de datos.
- Ingeniería inversa de bases de datos (desde sistemas gestores de bases de datos (DBMS) existentes a diagramas de Entidad-Relación).
- Generador de informes para generación de documentación.
- Editor de figuras.
- Alta velocidad a la hora de cargar y salvar los proyectos.
- Soporte multilenguaje. (Paradigm 2007)

2.7.3. Rational Rose Data Modeler.

Rational Rose Data Modeler es una herramienta de modelado visual que posibilita que los diseñadores de bases de datos, analistas, arquitectos, desarrolladores y todos los demás

miembros del equipo de desarrollo trabajen juntos, capturando y compartiendo los requerimientos de negocio y dándoles seguimiento a medida que cambian a través del proceso. Usa la notación UML para unificar a los diseñadores de bases de datos con el equipo de desarrollo de software. (Rational 2007)

Características de Rational Rose Data Modeler.

Integraciones IDE con:

- Borland JBuilder versiones 7.0 a 10.0.
- Sun Forte for Java Community and Enterprise Editions 3.0
- Microsoft Visual Studio 6.
- Microsoft Visual Studio 2003.
- Microsoft Visual Studio 2005 (Rational 2007)

Diagramas.

- Clases, Componentes, Despliegue, Secuencia, Caso de Uso.
- Colaboración.
- Almacenamiento Físico / Despliegue.
- Datos Físicos / Tablas. (Rational 2007)

2.8. ¿Por qué Visual Paradigm 6.4 For UML Enterprise Edition como herramienta CASE de modelado?

Para la realización de los diagramas se utilizará la herramienta Visual Paradigm 6.4, siguiendo la política establecida en la universidad para el desarrollo de sus productos. Esta herramienta permite de una manera muy fácil desarrollar los diagramas necesarios para desarrollar los servicios propuestos, y además el equipo de desarrollo presenta conocimientos básicos de la herramienta lo que posibilita un mejor desempeño con la misma.

2.9. Gestor de base de datos.

Los sistemas de gestión de bases de datos o SGBD (del inglés database management system, abreviado DBMS) son software específicos, que sirven de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. En la actualidad se puede encontrar en el mercado

SGBD libres, SGBD no libres y SGBD no libres y gratuitos. Por las características peculiares que presentan los SGBD libres es que se deciden usar estos en el desarrollo de los servicios propuestos. Entre los SGBD libres que existen actualmente se encuentran: MySQL, PostgreSQL, Firebird, SQLite, Apache Derby.

2.9.1. PostgreSQL 8.2.4.1.

PostgreSQL es un poderoso objeto de código abierto, sistema de base de datos relacional. Cuenta con una arquitectura probada que se ha ganado una sólida reputación de confiabilidad, integridad de datos y corrección. Funciona en todos los principales sistemas operativos, incluyendo Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), y Windows. Tiene soporte completo para claves foráneas, uniones, vistas, disparadores y procedimientos almacenados (en varios idiomas). Es compatible con el almacenamiento de objetos binarios, incluyendo imágenes, sonidos o vídeo. Tiene interfaces de programación nativa de C / C + +, Java. NET, Perl, Python, Ruby, Tcl, ODBC, entre otros. (PostgreSQL 1996)

Características de PostgreSQL 8.2.4.1.

- **Altamente extensible:** Soporta operadores, funcionales métodos de acceso y tipos de datos definidos por el usuario.
- **Soporte SQL comprensivo:** Soporta la especificación SQL99 e incluye características avanzadas tales como las uniones SQL92.
- **Integridad Referencial:** Garantiza la validez de los datos de la base de datos.
- **API Flexible:** La flexibilidad del API de PostgreSQL ha permitido a los vendedores proporcionar soporte al desarrollo fácilmente para el RDBMS PostgreSQL. Estas interfaces incluyen Object Pascal, Python, Perl, PHP, ODBC, Java/JDBC, Ruby, TCL, C/C++, y Pike.
- **Lenguajes Procedurales:** PostgreSQL tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL. Este lenguaje es comparable al lenguaje procedural de Oracle, PL/SQL. Otra ventaja de PostgreSQL es su habilidad para usar Perl, Python, o TCL como lenguaje procedural embebido.
- **Cliente/Servidor:** PostgreSQL usa una arquitectura proceso-por-usuario cliente/servidor. Esta es similar al método del Apache 1.3.x para manejar procesos. Hay

un proceso maestro que se ramifica para proporcionar conexiones adicionales para cada cliente que intente conectar a PostgreSQL. (PostgreSQL 1996)

2.9.2. MySQL.

Es un sistema de gestión de bases de datos relacional, fue creada por la empresa sueca MySQL AB, la cual tiene el copyright del código fuente del servidor SQL, así como también de la marca.(Angel 2010)

MySQL es un software de código abierto, licenciado bajo la GPL de la GNU. El lenguaje de programación que utiliza MySQL es SQL (del inglés Structured Query Language) que fue desarrollado por IBM y es utilizado de forma generalizada en las bases de datos relacionales.

Características de MySQL.

- Velocidad y robustez.
- Soporta gran cantidad de tipos de datos para las columnas.
- Gran portabilidad entre sistemas, puede trabajar en distintas plataformas y sistemas operativos.
- Cada base de datos cuenta con 3 archivos: Uno de estructura, uno de datos y uno de índice y soporta hasta 32 índices por tabla.
- Aprovecha la potencia de sistemas multiproceso, gracias a su implementación multihilo.
- Flexible sistema de contraseñas y gestión de usuarios, con un muy buen nivel de seguridad en los datos.

2.10. ¿Por qué PostgreSQL 8.2.4.1 como gestor de base de datos?

Por las características que presenta el gestor de base de datos PostgreSQL 8.2.4.1 y por las políticas que establece el Polo Petrosoft, fue que se decidió usar este en el desarrollo de los servicios de solicitud, diseminación selectiva y gestión de la información. Además, presenta otra característica interesante, no tiene costo incorporado a la licencia del software. PostgreSQL 8.2.4.1 es un software libre, liberado bajo la licencia BSD (del inglés Berkeley Software Distribution), lo que significa que cualquiera puede disponer de su código fuente, modificándolo a voluntad propia y distribuyéndolo libremente.

2.11. Lenguaje de Programación para servicios WEB.

Un lenguaje de programación es una técnica estándar de comunicación que permite expresar las instrucciones que han de ser ejecutadas en una computadora. (Lévez 2007)

2.11.1. PHP 5.2.4.

El lenguaje de programación PHP (del inglés Preprocessed Hypertext Pages, en lo adelante PHP) fue creado por Rasmus Lerdorf a finales de 1994, su principal objetivo está encaminado a la construcción de páginas web del lado del servidor.

PHP es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor, sin ninguna posibilidad de determinar que código ha producido el resultado recibido. PHP un lenguaje extremadamente simple para personas que no presenten un alto dominio del mismo, pero además, ofrece muchas características avanzadas para los programadores profesionales.

Es un lenguaje de código abierto y multiplataforma, puede ser utilizado en cualquiera de los principales sistemas operativos existentes, incluyendo Linux, Unix (incluido HP-UX, Solaris y OpenBSD), Microsoft Windows, Mac OS X, RISC OS y algunos más. Funciona tanto para Unix como para Windows de forma que el código creado para una de ellas no tiene por qué modificarse al pasar a la otra. Soporta diferentes servidores web como Apache, Microsoft Internet Information Server, Personal Web Server, iPlanet, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd entre otros. Permite utilizar programación orientada a objetos.

2.11.2. ASP.

Active Server Pages (en lo adelante ASP) es una solución de Microsoft basada en Visual Basic con las ventajas y desventajas que ello implica. La principal ventaja de ASP es que hay un flujo constante de trabajo para estos desarrolladores. Sin embargo, se debe tomar esta información con cautela pues las tendencias actuales pronostican un decremento de los servidores de Microsoft y un aumento en los sistemas Linux y BSD. Además, ASP es un sistema con nula portabilidad pues requiere necesariamente de un servidor Windows, con todas las implicaciones de alto costo, poca flexibilidad y escasa seguridad que estos equipos conllevan. (Mononeurona 2007)

2.11.3. JSP.

Java Server Pages (en lo adelante JSP) fue la respuesta de SUN (del inglés Stanford University Network) a las tecnologías script. JSP es la tecnología que más ha penetrado en el mundo corporativo. Ofrecen una gran versatilidad al momento de pensar en cómo resolver un problema, además, con JSP se aprenderá un lenguaje serio, altamente tipificado y que invita a desarrollar código bien estructurado y orientado a objetos. Las principales desventajas de JSP son la excesiva complejidad y aproximación elemental.

La excesiva complejidad deriva de enfocar cualquier problema como un EJB (del inglés Enterprise Java Bean), lo que muchas veces lleva a consumir grandes cantidades de tiempo y dinero. La aproximación elemental radica en que muchas empresas usan JSP como ASP, es decir, como una manera de hacer sus páginas dinámicas pero sin integrarlas a todos los módulos Java. (Mononeurona 2007)

2.12. ¿Por qué PHP 5.2.4 como lenguaje de programación?

Se utilizará PHP como lenguaje de programación por ser este un lenguaje de código abierto que brinda muchas ventajas, se integra con el servidor Apache y puede ser utilizado en diferentes sistemas operativos. Es fácil de aprender y de utilizar, brinda en su página oficial una gran cantidad de documentación y cuenta con una amplia biblioteca incluida de funciones, lo que hace más fácil el trabajo a desarrollar. Además, el CMS que se utilizará para el desarrollo de los servicios se encuentra implementado en PHP, lo que posibilita una mejor integración del trabajo realizado.

2.13. Entorno de Desarrollo Integrado (IDE).

Un entorno de desarrollo integrado (en lo adelante IDE) es un programa compuesto por varias herramientas que son utilizadas por los programadores para desarrollar código. Los IDE proveen entornos de trabajo amigables para diversos lenguajes de programación como C, C++, entre otros. Entre los IDE que más popularidad han alcanzado en la actualidad están Eclipse y NetBeans.

2.13.1. Eclipse.

Eclipse es un IDE desarrollado por Eclipse Foundation, de código abierto y multiplataforma. Sirve para muchos lenguajes de programación y permite la realización de múltiples operaciones integradas en proyectos.

Eclipse IDE provee plugins (aplicación informática que se relaciona con otra aplicación del mismo tipo para adicionarle una función o ventaja específica) para extender las capacidades del entorno de desarrollo.

Eclipse dispone de un editor de texto, la compilación es en tiempo real. Realiza el control de versiones con CVS, y permite añadir otros controladores de versión como Hibernate. Cuenta con asistentes Wizards para creación de proyectos, clases, tests, refactorización entre otras cosas.

2.13.2. NetBeans.

NetBeans es un IDE desarrollado por Sun Microsystems, de código abierto y multiplataforma. Permite diseñar aplicaciones de forma fácil con solo arrastrar objetos a la interfaz de un formulario. Es una plataforma pensada para escribir, compilar, depurar y ejecutar programas.

NetBeans no solo permite el desarrollo de aplicaciones de escritorio, también permite el desarrollo de aplicaciones para la web y para dispositivos portátiles. La programación en este IDE se realiza a través de componentes modulares o módulos. Las aplicaciones construidas a partir de módulos pueden ser extendidas ya que estos permiten ser desarrollados independientemente por otros desarrolladores de software, de ahí que sea una aplicación flexible/extensible.

Entre las características de la plataforma están la administración de las interfaces de usuario, administración de las configuraciones del usuario, administración del almacenamiento, administración de ventanas, frameworks basado en asistentes, entre otras cosas.

2.14. ¿Por qué NetBeans como IDE?

Eclipse es un entorno de desarrollo integrado que, aunque pretende ser un entorno versátil tolerando varios lenguajes de programación, con el que mejor se integra es con el lenguaje Java.

NetBeans es un entorno de desarrollo que tiene una interfaz amigable y fácil de comprender aún cuando los usuarios son inexpertos. Posee herramientas para crear aplicaciones

profesionales ya sean de escritorio, empresariales, web con PHP 5, móviles y aplicaciones SOA, no solo en Java sino también en C/C++ y Ruby. Incluye soporte para el trabajo con Symfony: un framework escrito en PHP. Provee soporte para el trabajo con XML, AJAX y modelado empleando UML.

Por las características que presenta NetBeans IDE y por ser además una plataforma flexible es que se determinó el uso de este IDE en el desarrollo de los servicios de solicitud, disseminación selectiva y gestión de la información del Polo Petrosoft.

2.15. Servidor Web.

Un servidor web sirve contenido estático a un navegador, carga un archivo y lo sirve a través de la red al navegador de un usuario. Este intercambio es mediado por el navegador y el servidor que hablan el uno con el otro mediante el protocolo HTTP (del inglés Hypertext Transfer Protocol, en lo adelante HTTP). Se pueden utilizar varias tecnologías en el servidor para aumentar su potencia más allá de su capacidad de entregar páginas HTML (del inglés HyperText Markup Language); éstas incluyen scripts CGI (del inglés Common Gateway Interface), seguridad SSL (del inglés Secure Sockets Layer) y páginas activas del servidor (ASP).

2.15.1. Apache 2.0.

Apache es el servidor web hecho por excelencia, su configuración, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. La licencia Apache es una descendiente de la licencias BSD, no es GPL (del inglés General Public License, en lo adelante GPL). Esta licencia permite hacer lo que quieras con el código fuente siempre que les reconozcas su trabajo. (Ciberaula 2006)

Ventajas del Servidor Apache.

- Compatibilidad con una gran cantidad de Sistemas Operativos, lo que lo hace prácticamente universal.
- Es una tecnología gratuita de código fuente abierto, esto brinda la posibilidad de ver lo que se está instalando en el servidor sin ocultar nada.

- Es un servidor altamente configurable de diseño modular. Otra cosa importante es que cualquiera que posea conocimientos en la programación de C o Perl puede escribir un módulo para realizar una función determinada.
- Apache trabaja con gran cantidad de lenguajes script como Perl, PHP, Java, paginas JSP y otros. Poseyendo todo el soporte necesario para tener páginas dinámicas.
- Apache permite personalizar las respuestas ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto (Ciberaula 2006)

2.16. ¿Por qué Apache como Servidor Web?

Se utilizará Apache como servidor Web porque es un servidor multiplataforma que permite ser adaptado a las necesidades del cliente. Permite la creación de registros controlando así lo que sucede en el servidor. Es un servidor compatible con el sistema de gestión de contenidos Drupal, lo que posibilita un mejor desarrollo de los servicios de gestión de información, debido a que la Intranet a la cual serán incorporados los mismos utiliza y aprovecha las ventajas de este servidor.

2.17. Sistema de Gestión de Contenidos (CMS).

Un sistema de gestión de contenidos (en lo adelante CMS) es un software que permite manejar cada pieza de contenido en un sitio web. El contenido puede ser un simple texto, música, documentos u otro cualquier aspecto.

2.17.1. Joomla.

Es un sistema de administración de contenidos de código abierto construido con PHP bajo una licencia GPL. Este administrador de contenidos se usa para publicar en Internet e Intranets utilizando una base de datos MySQL. (Ferrer 2007)

Características Generales del CMS Joomla.

- **Automatización en la publicación:** Los documentos pueden programarse para que sean publicados en una fecha determinada y lo mismo a la inversa.
- **Archivo e historial:** No hay necesidad de borrar las páginas o publicaciones antiguas gracias a que brinda la posibilidad de ser almacenados para futuras consultas.

- **Formatos de lectura:** Genera los documentos en formato PDF, en versión imprimible, y en XML (del inglés Extensible Markup Language).
- **Envío por correo:** Permite enviar los documentos publicados por correo.
- **Valoración de contenidos:** Permite que los usuarios emitan su criterio acerca del sitio y del contenido del mismo.
- **Comentarios:** Los usuarios pueden emitir sus criterios en la página de contenidos.

2.17.2. Drupal 6.9.

Es un CMS (sistema de gestión de contenidos, en lo adelante CMS) modular y muy configurable. Es un programa de código abierto, con licencia GNU/GPL, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. Se 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. No obstante, su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales disponibles, hace que sea adecuado para realizar diferentes tipos de sitios web. (Reyero 2006)

Características Generales de Drupal 6.9.

- **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.
- **Módulos:** Presenta diferentes módulos con funcionalidades como página de categorías, autenticación mediante jabber, mensajes privados.
- **Personalización:** Permite personalizar el entorno de trabajo según las preferencias de los usuarios.
- **Gestión de usuarios:** Permite la autenticación de usuarios posibilitando además agrupar los mismos según el rol que desempeñen.
- **Gestión de contenido:** Posee las ventajas que tienen los software controladores de versiones permitiendo seguir las actualizaciones del contenido. Permite mantener comentarios sobre los sucesivos cambios o deshacer los cambios recuperando una versión anterior.

- **Plantillas:** 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 con PHP.
- **Sindicación del contenido:** Drupal exporta el contenido en formato RDF/RSS para ser utilizado por otros sitios web. Esto permite que cualquiera con un agregado de noticias, tal como NetNewsWire o Radio UserLand visualice el contenido publicado en la web desde el escritorio. (Reyero 2006)
- **Plataforma:** Drupal permite la independencia de las bases de datos, este incorpora una capa de abstracción de base de datos actualmente implementada y mantenida para MySQL y PostgreSQL, aunque permite además incorporar soporte para otras.
- **Administración y Análisis:** La administración y configuración del sistema se puede realizar enteramente con un navegador y no precisa de ningún software adicional. Toda la actividad y los sucesos del sistema son capturados en un registro de eventos, que puede ser visualizado por un administrador. (Reyero 2006)
- **Características de comunidad:** Proporciona la participación en foros relacionados con comentarios publicados y permite entre otras cosas crear encuestas online configurables.
- **Rendimiento y escalabilidad:** Controla la congestión, incorporando mecanismos configurables y ajustables que habilitan y deshabilitan módulos diseñados para esta función.

2.18. ¿Por qué Drupal 6.9 como CMS?

Se utilizará como CMS Drupal, por las características que presenta, aprovechando que es un sistema de código abierto multiplataforma, compatible con servidores como Apache o Microsoft IIS y sistemas operativos como Linux, Solaris, Windows y Mac OSX. Es un CMS que permite la escalabilidad en los sistemas desarrollados con él para ajustarse a futuras necesidades con módulos. Además, al ser un CMS implementado en PHP, es totalmente portable. Presenta múltiples idiomas brindando la posibilidad de crear portales multilingües, posee canales de sindicación y entre otras cosas permite imprimir el contenido de las páginas creadas así como el envío de estas por correo electrónico. Por último, y no menos importante, los servicios que serán desarrollados serán incorporados en la intranet del Polo Petrosoft la cual presenta como CMS Drupal 6.9.

2.19. CONCLUSIONES.

La buena elección de la metodología así como de las herramientas a utilizar facilita que el desarrollo del trabajo sea eficiente. En la actualidad existen diferentes metodologías, las cuales se utilizan según el tipo de proyecto a desarrollar y se cuenta con muchas herramientas que hacen más fácil y rápido el trabajo. La idea principal es utilizar las herramientas y las metodologías adecuadas en la situación adecuada, pues el éxito depende del buen uso que se haga de estas.

CAPÍTULO III: PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

3.1 INTRODUCCIÓN.

Este capítulo tendrá como objetivo comprender y detallar de forma sencilla el entorno donde está enmarcado el problema. La elaboración de un glosario para definir un vocabulario común entre todas las personas que interactúan en el negocio y conseguir un mejor entendimiento del mismo. Especificar las reglas a cumplir en el negocio para un uso correcto de la aplicación, definir los requisitos funcionales y no funcionales que deben cumplir los servicios de solicitud, diseminación selectiva y gestión de información.

3.2 Modelamiento del Negocio.

- El negocio no es más que el ambiente o entorno en que se enmarca el problema a resolver.
- El Modelamiento del Negocio es la primera fase por la que atraviesa un producto antes de ser construido.

3.3 Descripción del negocio.

El Polo Productivo Petrosoft de la Facultad 9 se encuentra vinculado directamente a la producción de software relacionados con el Petróleo. En el Polo existe un evento que es el concerniente al desarrollo de Talleres Científicos, en el cual los estudiantes presentan trabajos investigativos.

Primeramente, para presentar trabajos en el taller los estudiantes tienen que enviar la información por correo electrónico al Especialista en Gestión del Conocimiento. El Especialista revisa el trabajo para comprobar que no contenga errores, si el trabajo no contiene errores es adicionado al cronograma de trabajos a exponer y se le notifica al estudiante la hora en que debe presentarlo. En caso contrario no se presenta el trabajo en el evento.

Como el flujo de actividades desarrolladas en el Polo es dinámico, para mantenerse actualizados en todos los eventos que se realizan y en cuanto a otras novedades de interés, el Polo se propuso, cada 15 días, confeccionar y enviar un Boletín Electrónico. Este boletín se envía por correo electrónico a todos los profesores del Polo. Antes de ser enviado, el Jefe de Grupo de Desarrollo Humano revisa dicho boletín, si está realizado correctamente lo envía al Jefe del Polo y este lo distribuye a los profesores, en caso contrario notifica a los responsables y estos corrigen

las deficiencias que presenta. Una vez corregidas, se vuelve a enviar al Jefe de Grupo de Desarrollo Humano.

La confección del boletín suele en ocasiones tornarse un proceso algo tedioso, ya que existen deficiencias pues no se cuenta con un software especializado en la lectura de contenidos de otras páginas. Para mantener actualizado el boletín hay que pasarse largas jornadas de trabajo realizando búsquedas en Internet. Estas búsquedas las realizan los estudiantes que integran el Grupo de Gestión de la Información del Polo. Cuando se realizan estas búsquedas si la información encontrada es actual se incorpora al boletín. En caso contrario se analiza si es relevante para determinar entonces si se incorpora.

Los Jefes de Grupo del Polo cuentan con el privilegio de solicitar ediciones antiguas de los boletines, pero también pueden solicitar información al Grupo de Gestión de Información y recibir actualizaciones de la misma. Estos procesos se realizan de forma manual, existiendo estudiantes encargados de dar respuesta a dichas solicitudes. Cuando se le ha dado respuesta a las solicitudes se envía la información a la persona que la solicita, en caso de que no hayan sido realizadas la mismas se notifica de igual manera.

3.4 Objeto de Automatización.

Teniendo en cuenta el negocio descrito anteriormente se puede decir que se desean automatizar varios servicios que ayudarán a mejorar las dificultades antes planteadas. Por tanto, se propone primeramente automatizar el servicio de solicitud de información por ser el que presenta más problemas, ya que en muchos casos la misma no se entrega en el tiempo estipulado, debido a que las personas encargadas presentan dificultades para realizar la misma, o los funcionarios que tienen que solicitar información no lo hacen en el tiempo establecido porque se encuentran realizando actividades cuando se les va a recoger la solicitud. Se propone informatizar este servicio y el de diseminación selectiva realizando un formulario en forma digital para recopilar los datos de la información a solicitar.

Seguidamente se automatizará el servicio de envío de información relacionada con eventos científicos para que no se realice por medio del correo electrónico como actualmente se hace, sino que exista la posibilidad de que los mismos estudiantes suban la información a la intranet del Polo.

Así también pasará con el boletín del Polo, este será incorporado a la intranet del Polo y no habrá necesidad de que se envíe por correo, evitando que no llegue a su destino en caso de que haya congestión de las redes o los destinatarios presenten problemas con el correo. Además, se

incorporará a la Intranet un canal RSS (del inglés Really Simple Syndication) para que diariamente se reciban noticias relevantes y actuales relacionadas con temas de interés para el Polo.

Propuesta de la aplicación.

Con el objetivo de mejorar los servicios anteriormente planteados para un mejor desarrollo de las tareas realizadas en el Polo Petrosoft, se propone automatizar los servicios antes expuestos e incorporarlos en la Intranet del Polo Petrosoft de la Facultad 9.

3.5 Actores, Trabajadores y entidades del negocio.

Actor no es más que el rol que alguien o algo toma cuando interactúa con el negocio. Los actores representan un rol y no una persona específica. Ejemplo de esto pueden ser: Clientes, Propietarios e incluso la Organización para la que produce el software.

3.6.1. Definición de los actores del negocio.

Actores del negocio	Descripción
Usuario (Jefe del Grupo de Aplicación, Jefe del Grupo de Asesoramiento Técnico a proyectos, Jefe del Grupo de Procesamiento de Datos, Jefe del Grupo de Interpretación).	Es el que solicita información al Jefe del Grupo de Desarrollo Humano.
Estudiante	Es el que envía el trabajo para presentarlo en el Taller del Polo.
Profesor	Recibe el boletín del Polo.

Tabla 3: Definición de actores del negocio

CAPÍTULO III: PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

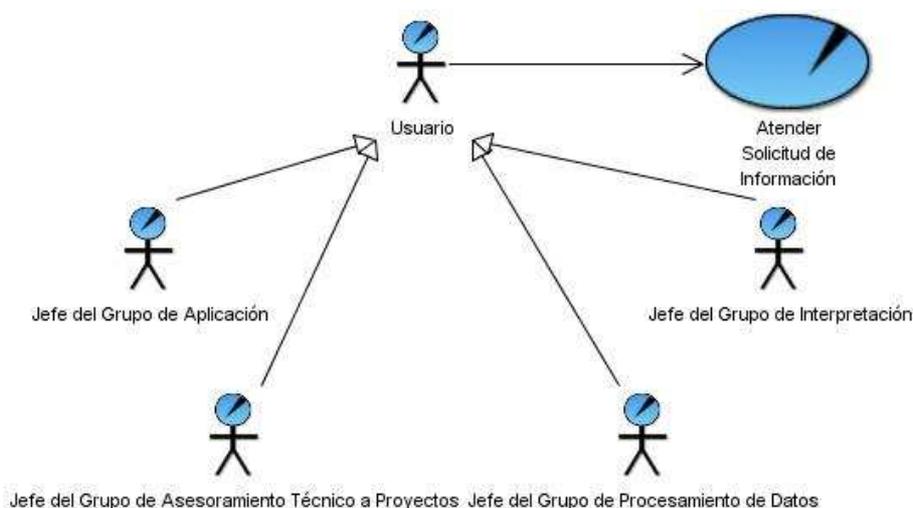


Ilustración 1: Diagrama de caso de uso del negocio: Atender Solicitud de Información.

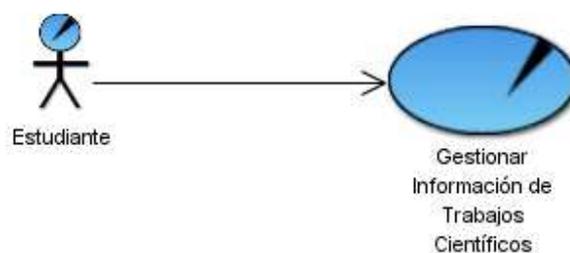


Ilustración 2: Diagrama de caso de uso del negocio: Recibir Información de Trabajos Científicos.

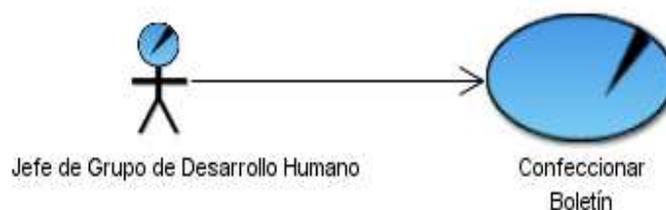


Ilustración 3: Diagrama de caso de uso del negocio: Confeccionar Boletín.

Trabajadores del negocio: Serán aquellos que trabajan en el negocio. Muestran las responsabilidades que puede tener una persona, por lo que representa una abstracción de una persona que actúa dentro del negocio. (Pressman 2001)

3.6.2. Definición de los trabajadores del negocio.

Trabajadores del negocio	Descripción
Jefe de Grupo de Desarrollo Humano.	Es el encargado de enviar la documentación a los jefes de los proyectos del Polo y de facilitar la información para confeccionar el boletín, revisarlo y enviárselo al Jefe de Polo.
Grupo de Gestión de Información.	Es el encargado de atender las solicitudes de información, así como de recopilar otros datos de interés para adicionarle al boletín y confeccionarlo.
Jefe de Polo.	Es el encargado de enviar el boletín a los integrantes del Polo.
Especialista en Gestión del Conocimiento.	Es el encargado de recepcionar la información relacionada con los Talleres Científicos del Polo y de incluir los trabajos en el cronograma de presentaciones.

Tabla 4: Definición de trabajadores del negocio.

Entidades: Las entidades no son más que las clases que van a guardar la información perdurable en el tiempo.

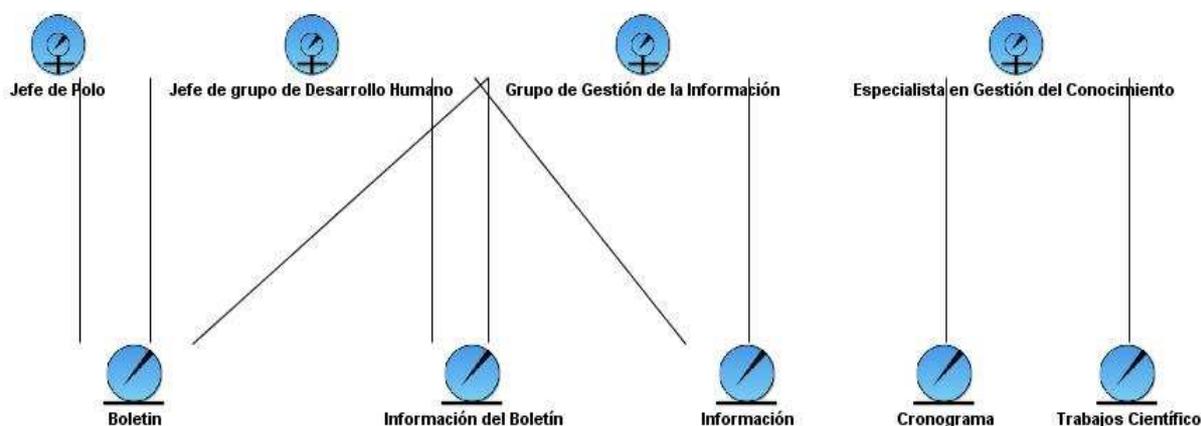


Ilustración 4: Modelo de objetos

3.7. Descripción textual de los casos de uso del negocio.

3.7.1. Descripción del caso de uso Atender Solicitud de Información.

Nombre del Caso de Uso	Atender Solicitud de Información.
Actores	Usuario (inicializa).
Propósito	Atender la solicitud de información realizada por los jefes de proyectos.
Resumen: El caso de uso se inicializa cuando los jefes de grupo solicitan información al Jefe de grupo de desarrollo humano. El jefe de grupo de desarrollo humano informa a los estudiantes encargados realizar las búsquedas de información. Cuando la búsqueda se ha efectuado los estudiantes le informan al jefe de grupo de desarrollo humano y este envía la información solicitada a las personas correspondientes.	
Caso de uso asociados	
Flujo Básico	
Acción del Actor	Respuesta
1. El usuario (jefes de grupo del Polo Petrosoft) solicita información al jefe de grupo de desarrollo humano.	2. El jefe de grupo de desarrollo humano envía a los estudiantes encargados de realizar esta actividad a atender la solicitud.
	3. Los estudiantes entrevistan al usuario para conocer cuál es la necesidad de información que presentan.
	4. Los estudiantes buscan la información solicitada y se la envían al jefe de grupo de desarrollo humano. El jefe de grupo de desarrollo humano envía la información al usuario.

Prioridad	
Mejoras	
Flujo Alternativo	
Si no se encontró información relacionada con la solicitud se le notifica al jefe de grupo de desarrollo humano y este le notifica al usuario.	

Tabla 5: Descripción textual del caso de uso Atender Solicitud de Información

3.7.2. Descripción del caso de uso Recibir Información de Trabajos Científicos.

Nombre del Caso de Uso	Gestionar Información de Trabajos Científicos.
Actores	Estudiante (inicializa).
Propósito	Recibir la información para presentar en los talleres científicos.
Resumen: El caso de uso se inicializa cuando el estudiante envía por correo electrónico la información a presentar en el taller del Polo al Especialista en Gestión del Conocimiento. El Especialista recepciona la información y la revisa, si está correcta la incluye en el cronograma de presentaciones y le notifican al estudiante la hora de exposición. En caso contrario no se incluye en el cronograma de presentaciones.	
Caso de uso asociados	
Flujo Básico	
Acción del Actor	Respuesta
1. El estudiante envía al Especialista de Gestión del Conocimiento el trabajo a presentar en el taller científico.	2. El Especialista revisa la información enviada por el estudiante.
	3. Si la información esta correcta incluye el trabajo en el cronograma de actividades.

CAPÍTULO III: PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

	4. El Especialista notifica al estudiante la hora de exposición del trabajo.
Prioridad	
Mejoras	
Flujo Alterno	
Si la información no está correcta no se incluye el trabajo en el cronograma de presentaciones y se le notifica al estudiante.	

Tabla 6: Descripción textual del caso de uso Recibir Información de Trabajos Científicos

3.7.3. Descripción del caso de uso Confeccionar Boletín.

Nombre del Caso de Uso	Confeccionar Boletín
Actores	Jefe de Grupo de Desarrollo Humano (inicializa).
Propósito	Confeccionar el boletín del Polo Petrosoft.
Resumen: El caso de uso se inicia cuando Jefe de Grupo de Desarrollo Humano indica que hay que confeccionar el boletín y envía la información necesaria para su confección. El Grupo de Gestión de Información recopila más información de interés y elaboran el mismo. Cuando el boletín es confeccionado es revisado por el Jefe de Grupo de Desarrollo Humano y luego se lo envía al Jefe de Polo, que es el encargado de distribuirlo a todos los profesores.	
Caso de uso asociados	
Flujo Básico	
Acción del Actor	Respuesta
1. El Jefe de Grupo de Desarrollo Humano indica que hay que confeccionar el boletín.	2. El Jefe de Grupo de Desarrollo Humano envía la información necesaria al Grupo de Gestión de la Información para confeccionar el boletín.

	3. El Grupo de Gestión de Información recopila más información de interés para añadirla al boletín.
	4. El Grupo de Gestión de Información elabora el boletín y se lo envía al Jefe de Grupo de Desarrollo Humano para que este sea revisado.
	5. Si el boletín tiene el formato correcto, se le envía al Jefe de Polo y este distribuye dicho boletín.
Prioridad	
Mejoras	
Flujo Alternativo	
Si el boletín no tiene el formato correcto o presenta errores, se le envía nuevamente a los integrantes del Grupo de Gestión de Información y estos lo corrigen.	

Tabla 7: Descripción textual del caso de uso Confeccionar Boletín.

3.8. Reglas del Negocio.

Las reglas del negocio permiten definir los límites lógicos de las aplicaciones. En ellas las aplicaciones reflejan las restricciones que existen, evitando la realización de acciones no válidas. (Rodríguez Quintana 2006)

Las reglas de negocios concretan y controlan la estructura, la actividad y la estrategia de una organización. Pueden estar definidas en manuales de procedimiento, contratos, o pueden existir como conocimiento o experiencia que tienen los empleados. Las reglas de negocios son dinámicas, están sujetas a cambios en el tiempo y pueden encontrarse en todo tipo de aplicaciones.

Para el adecuado funcionamiento de los servicios a desarrollar se deben cumplir las siguientes reglas:

Solicitud de Información:

- Las solicitudes serán realizadas con al menos 1 día de antelación.

- Grupo de Gestión de la Información son los encargados de revisar las solicitudes almacenadas.
- Los integrantes del Grupo de Gestión de la Información son los encargados de dar respuesta a las solicitudes.
- Los Jefes de Grupo no pueden realizar más de 10 solicitudes de información en 1 mes.

Presentar Trabajos Científicos:

- Los trabajos deben tener un alto rigor científico y contar con un problema a resolver, un campo de acción, objeto de estudio.
- El contenido de los trabajos debe proceder de una fuente de información confiable, no puede proceder de Wikipedias o de Blogs procedentes de Internet.
- Los trabajos tienen que ser revisados antes de ser presentados en el taller.
- Los trabajos científicos que se suban a la aplicación tienen que ser documentos Word.
- Los trabajos científicos tienen que ser subidos con el formato: grupo_nombre y apellidos.
- El nombre de los trabajos no puede contener acentos.

3.9. Levantamiento de Requisitos.

En el desarrollo de software la fase de Requerimientos es una de las más importantes, pues de la correcta realización de la misma es que se obtiene un producto con todas las funcionalidades deseadas por el cliente.

3.9.1. Requisitos Funcionales.

Los requisitos funcionales no son más que las acciones o funcionalidades que el sistema tiene que realizar.

El sistema debe permitir:

RF1: Crear nuevas solicitudes de información científica.

RF2: Editar solicitudes de información científica en no más de 2 días a los usuarios que las realizan.

RF3: Cancelar solicitudes de información científica.

RF4: Eliminar solicitudes de información científica.

RF5: Atender solicitudes de información científica.

RF6: Mostrar información.

RF7: Subir trabajos científicos a la aplicación.

RF8: Visualizar trabajos científicos.

RF9: Aceptar trabajos científicos.

RF10: Rechazar trabajos científicos que contengan errores.

RF11: Enviar notificación informando si el trabajo es aprobado o no para ser presentado en el Taller Científico del Polo.

RF12: Confeccionar cronograma de presentaciones con los trabajos aprobados del Taller Científico del Polo.

RF13: Publicar fecha del evento.

RF14: Publicar cronograma.

RF15: Activar subida de trabajos científicos a la aplicación.

RF16: Visualizar Cronograma de presentaciones.

RF17: Visualizar boletín del Polo en formato HTML en el Portal del Polo.

RF18: Diseminar información científica a los Jefes de Grupo del Polo.

3.9.2. Requisitos no Funcionales.

Los requisitos no funcionales son cualidades o características que el sistema debe poseer.

RNF1: Apariencia o Interfaz.

RNF1.1: Los colores a emplear en el sistema deben estar relacionados con la entidad en la cual se implantará el sistema.

RNF2: Rendimiento.

RNF2.1: Se debe garantizar que el tiempo de respuesta del sistema a las solicitudes se produzca en la mayor brevedad posible (menos de 5 segundos).

RNF2.2: El sistema debe presentar pocas imágenes en las páginas para garantizar una respuesta más rápida (como máximo 3 imágenes o menos por página).

RNF3: Seguridad.

RNF3.1: Solo las personas autorizadas pueden tener acceso a la información: Las únicas personas autorizadas para ver las solicitudes de información almacenadas son el Jefe de Grupo de Desarrollo Humano y los integrantes del Grupo de Gestión de la Información.

RNF3.2: El sistema debe tener un mecanismo de protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.

RNF4: Disponibilidad.

RNF4.1: El sistema debe permitir a los usuarios conectarse desde cualquier computadora que esté conectada a la red dentro de la universidad.

RNF4.2: El sistema no debe tener más de 1 hora de no disponibilidad en el mes.

RNF5: Portabilidad.

RNF5.1: El sistema tiene que ser multiplataforma.

RNF5.2: El sistema podrá ser visualizado en sistemas operativos superiores a Windows 98, Linux.

RNF7: Hardware.

Servidores

RNF7.1: PC servidor con 256 MB(o superior) de memoria RAM.

Clientes

RNF7.2: PC clientes con 128MB de memoria RAM como mínimo.

FNR8: Software del Sistema.

Servidores.

RNF8.1. Sistema operativo Microsoft Windows XP.

RNF8.2: Servidor Web Apache 2.0.

Clientes.

RNF8.4: Sistema operativo Microsoft Windows XP o Linux.

Internet.

RNF8.5: Mozilla Firefox 1.5(o superior).

3.10. Patrones de Casos de Uso.

Los patrones de casos de uso son parejas de problemas que codifica o estandariza buenos principios y sugerencias relacionados frecuentemente con la asignación de responsabilidades. El uso de patrones posibilita que el sistema logre una mayor reusabilidad y que se obtengan mejores resultados en la confección del mismo.

Los patrones que serán utilizados en la implementación de los servicios de gestión de información son:

- Reglas del Negocio: Se utiliza para extraer la información de las funcionalidades del negocio a partir de las reglas definidas para este.
- Roles Comunes: Este patrón se emplea cuando existen varios actores que jueguen un mismo rol.
- CRUD: Este patrón permite crear, leer, actualizar, eliminar, es decir modela las funciones básicas en las bases de datos y casos de usos.

- Extender: Patrón que se utiliza para agregar otras funcionalidades a los casos de uso base. Su uso es opcional, o sea no siempre se realizan las acciones de los casos de uso extendidos.

3.11. Descripción del Sistema Propuesto.

Un actor del sistema es una idealización de una persona externa, de un proceso, o de una cosa que interactúa con un sistema, un subsistema, o una clase. Un actor caracteriza las interacciones que los usuarios exteriores pueden tener con el sistema. En tiempo de ejecución, un usuario físico puede estar limitado a los actores múltiples dentro del sistema. Diferentes usuarios pueden estar ligados al mismo actor y por lo tanto pueden representar casos múltiples de la misma definición de actor. (Hernández 2009)

Un actor del sistema es un trabajador del negocio que va a realizar actividades que serán automatizadas.

3.11.1. Descripción de los Actores del Sistema.

Actor	Descripción
Usuario	Es el encargado de generalizar todos los usuarios del sistema, subir la información para presentar en el Taller del Polo, visualizar el boletín y visualizar el cronograma de presentaciones.
Grupo de Gestión de Información (GGI).	Es el encargado de gestionar el listado de solicitudes de información.
Especialista en Gestión del Conocimiento (EGC).	Es el encargado de gestionar el listado de los trabajos científicos, gestionar el cronograma de presentaciones y gestionar los eventos.
Jefe de Grupo del Polo (Jefe del Grupo de Procesamiento de Datos, Jefe del Grupo de Aplicación, Jefe del Grupo de Asesoramiento a Proyectos, Jefe del Grupo de Interpretación).	Es el encargado de gestionar las solicitudes de información científica, solicitar información científica y recibir información selectiva.

Tabla 8: Definición de Actores del Sistema.

3.12. Diagrama de casos de uso del sistema.

Un diagrama de Casos de Uso del Sistema describe parte del modelo de casos de uso y muestra un conjunto de casos de uso y actores con una asociación entre cada par actor/caso de uso que interactúan. (Hernández 2009).

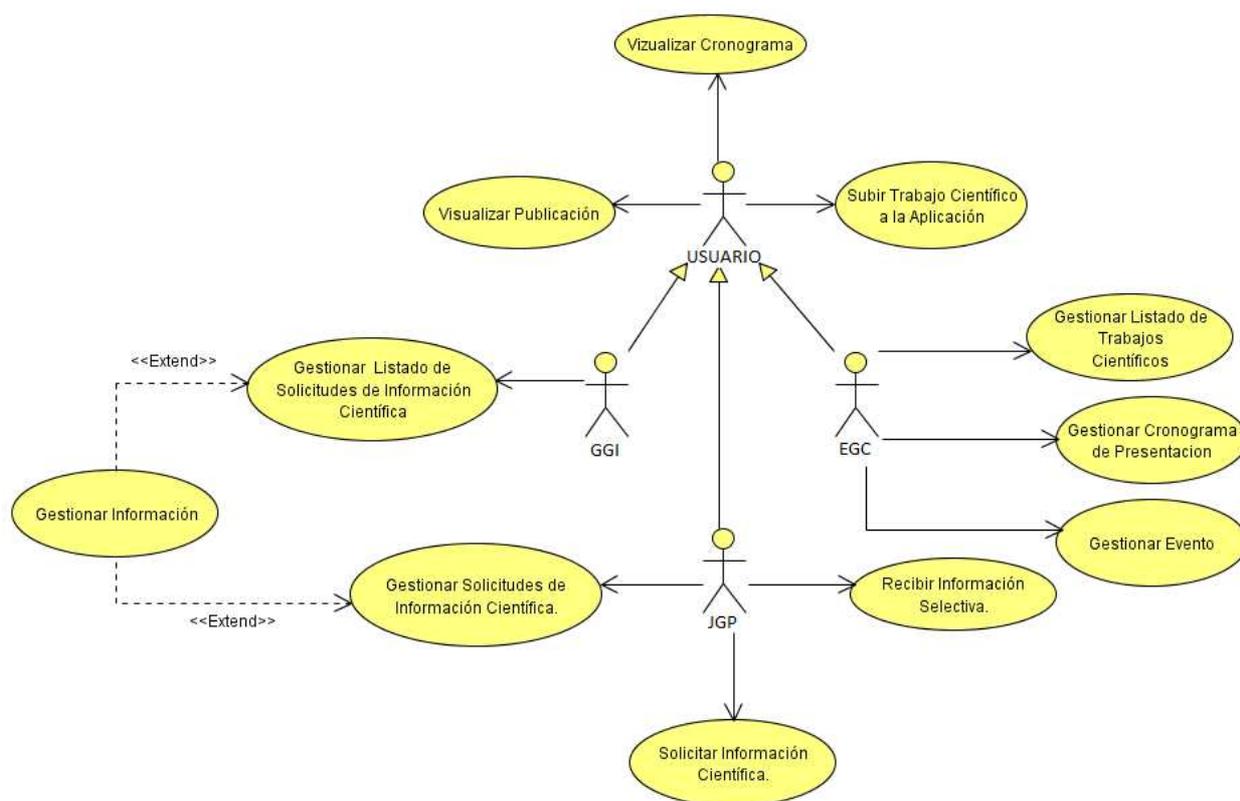


Ilustración 5: Diagrama de Casos de Uso de Sistema.

3.13. Descripción de los Casos de Uso del Sistema.

3.13.1. Descripción del Caso de Uso Solicitar Información Científica (Ver Anexo 1).

Caso de Uso	Solicitar Información Científica.
Actores	Jefes de Grupo del Polo (JGP).

Descripción: Este caso de uso permite al usuario crear nuevas solicitudes de información científica.	
Referencias	RF1.

Tabla 9: Descripción del Caso de Uso Solicitar Información Científica.

3.13.2. Descripción del Caso de Uso Gestionar Solicitudes de Información Científica (Ver Anexo 2).

Caso de Uso	Gestionar Solicitudes de Información Científica.
Actores	Jefes de Grupo del Polo (JGP).
Descripción: Este caso de uso permite al usuario mostrar, editar, eliminar, y cancelar las solicitudes realizadas por él.	
Referencias	RF2, RF3.

Tabla 10: Descripción del caso de Uso Gestionar Solicitudes de Información Científica.

3.13.3. Descripción del Caso de Uso Gestionar Listado de Solicitudes de Información Científica (Ver Anexo 3).

Caso de Uso	Gestionar Listado de Solicitudes de Información Científica.
Actores	Grupo de Gestión de la Información(GGI)
Descripción: Este caso de uso permite al GGI gestionar las solicitudes de información existentes.	
Referencias	RF5.

Tabla 11: Descripción del Caso de Uso Gestionar Listado de Solicitudes de Información Científica.

3.13.4. Descripción del Caso de Uso Gestionar Información (Ver Anexo 4).

Caso de Uso	Gestionar Información.
Actores	
Descripción: Este caso de uso permite al sistema mostrar y eliminar información cuando el usuario seleccione la opción correspondiente.	
Referencias	RF4,RF6

CAPÍTULO III: PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Tabla 12: Descripción del Caso de Uso Gestionar Información.

3.13.5. Descripción del Caso de Uso Subir Trabajo Científico a la Aplicación (Ver Anexo 5).

Caso de Uso	Subir Trabajo Científico a la aplicación.
Actores	Usuario.
Descripción: Este caso de uso permite que los usuarios suban a la aplicación los trabajos científicos a presentar en el taller.	
Referencias	RF7

Tabla 13: Descripción del Caso de Uso Subir Trabajo Científico a la Aplicación.

3.13.6. Descripción del Caso de Uso Gestionar Listado de Trabajos Científicos. (Ver Anexo 6).

Caso de Uso	Gestionar Listado de Trabajos Científicos.
Actores	Especialista en Gestión del Conocimiento (EGC).
Descripción: Este caso de uso permite al Especialista gestionar la información relacionada con los trabajos científicos.	
Referencias	RF8, RF9, RF10, RF11.

Tabla 14: Descripción del Caso de Uso Gestionar Listado de Trabajos Científicos.

3.13.7. Descripción del Caso de Uso Confeccionar Cronograma de Presentaciones. (Ver Anexo 7).

Caso de Uso	Gestionar Cronograma de Presentaciones.
Actores	Especialista en Gestión del Conocimiento (EGC).
Descripción: Este caso de uso permite al Especialista gestionar la información relacionada con la presentación de los trabajos científicos.	
Referencias	RF12.

CAPÍTULO III: PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA

Tabla 15: Descripción del Caso de Uso Gestionar Cronograma de Presentaciones.

3.13.8. Descripción del Caso de Uso Gestionar Evento. (Ver Anexo 8).

Caso de Uso	Gestionar Evento.
Actores	Especialista en Gestión del Conocimiento (EGC).
Descripción: Este caso de uso permite al Especialista gestionar la información relacionada con el Taller Científico del Polo, la publicación del cronograma y la subida de los trabajos a la aplicación.	
Referencias	RF13, RF14, RF15.

Tabla 16: Descripción del Caso de Uso Gestionar Evento.

3.13.9. Descripción del Caso de Uso Visualizar Cronograma de Presentaciones. (Ver Anexo 9).

Caso de Uso	Visualizar Cronograma de Presentaciones.
Actores	Usuario.
Descripción: Este caso de uso permite al usuario ver el cronograma de presentaciones de los trabajos que serán presentados en el Taller del Polo.	
Referencias	RF16.

Tabla 17: Descripción del Caso de Uso Visualizar Cronograma de Presentaciones.

3.13.10. Descripción del Caso de Uso Visualizar Publicación (Ver Anexo 10).

Caso de Uso	Visualizar Publicación.
Actores	Usuario.
Descripción: Este caso de uso permite al usuario visualizar el boletín en el Portal del Polo.	
Referencias	RF17.

Tabla 18: Descripción del Caso de Uso Visualizar Publicación.

3.13.11. Descripción del Caso de Uso Recibir Información Selectiva (Ver Anexo 11).

Caso de Uso	Recibir Información Selectiva.
Actores	Jefe de Grupo del Polo (JGP).
Descripción: Este caso de uso permite los Jefes de Grupo del Polo recibir con determinada frecuencia información del tipo que la haya solicitado.	
Referencias	RF18.

Tabla 19: Descripción del Caso de Uso Recibir Información Selectiva.

3.14. CONCLUSIONES.

El desarrollo de este capítulo ha logrado un mejor entendimiento de los servicios a automatizar, así como las restricciones que deben cumplir estos para cumplir con los requisitos establecidos por el cliente.

Después de realizar un estudio detallado del negocio se procedió al modelamiento del mismo para un mejor entendimiento, se confeccionaron los diagramas de las principales actividades realizadas en el negocio y un glosario de términos comunes para todas las personas que interactúan en el negocio.

Se definieron los requisitos funcionales y no funcionales que los servicios deben presentar, se realizó el modelamiento del sistema para tener una visión más clara de quienes son los actores del mismo y las funciones que desarrollarán y se describieron de forma detallada las principales actividades.

Todo esto permitió entender de mejor forma las actividades fundamentales, tanto del negocio como de los servicios que deben ser automatizados, mostrando una visión más clara de cuáles serán las funcionalidades que debe presentar.

CAPÍTULO IV: IMPLEMENTACIÓN DE LA SOLUCIÓN PROPUESTA.

4.1 INTRODUCCIÓN.

En este capítulo se construye la propuesta de solución para los servicios de solicitud, disseminación selectiva y gestión de la información del Polo Petrosoft. Además, se incluye el modelo de diseño del sistema a partir de la arquitectura definida para el mismo, el modelo de datos y el modelo de despliegue.

4.2 Arquitectura.

La arquitectura de software es la estructura más detallada del sistema que incluye los componentes fundamentales del mismo, y que proporciona un marco de referencia para guiar, de manera más organizada la construcción del software entre los analistas, diseñadores, programadores y demás miembros del equipo de desarrollo. Entre sus objetivos fundamentales se encuentran definir los módulos principales del sistema, la responsabilidad que tendrá cada uno de los módulos, la interacción entre los mismos y controlar el flujo de los datos en el sistema.

Cada sistema posee una arquitectura propia y esta es definida según el tipo, aunque diferentes sistemas pueden presentar una misma arquitectura, la diferencia radica en las funcionalidades que realizará cada uno. Las arquitecturas más aceptadas y usadas en la actualidad son la arquitectura monolítica, cliente-servidor y tres capas.

Los CMS poseen una arquitectura que difiere de los demás sistemas, ya que estos proveen una estructura de soporte para el desarrollo, principalmente, en páginas web.

4.2.1 Arquitectura del CMS Drupal.

Drupal es un CMS que no se ajusta a un patrón de arquitectura específico. Posee una estructura propia, esta estructura es la que lo hace ser un CMS flexible que permite crear sitios web de forma fácil. La arquitectura de Drupal es una arquitectura modular, este tipo de arquitectura le permite otorgarle diferentes funcionalidades a las aplicaciones mediante la configuración de los módulos que trae este CMS por defecto o, incorporándole módulos sin afectar los que hay ya creados.

CAPÍTULO IV: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Drupal no considera los sitios web como un conjunto de páginas interrelacionadas entre sí, sino que estructura el contenido en una serie de elementos básicos que son: nodos, módulos, bloques y menús, permisos de usuario y plantillas. Cada uno de estos elementos tiene funciones específicas. A continuación se explican cada una de estas.

- **Nodos:** Elementos básicos en los cuales se almacena la información. A medida que el sitio web crece aumenta el número de nodos.
- **Módulos:** Elementos que operan sobre los nodos y otorgan funcionalidad a Drupal permitiendo incrementar sus capacidades o adaptarlas a las necesidades de cada usuario.
- **Menús:** Los menús permiten organizar los contenidos en las páginas web. Son los elementos que permiten acceder al usuario a la salida generada y procesada por los módulos a partir de la información almacenada en estos.
- **Permisos y control de Usuarios:** Permite especificar las tareas que puede realizar cada tipo de usuario en el sistema. Las operaciones que se pueden realizar sobre elementos situados en las capas inferiores se encuentran limitadas por las capas de control de usuarios y permisos.
- **Plantillas:** Establece la apariencia gráfica o estilo de la información que se muestra al usuario. Esta capa ofrece la ventaja de cambiar el diseño del sitio sin necesidad de modificar los contenidos de las páginas.

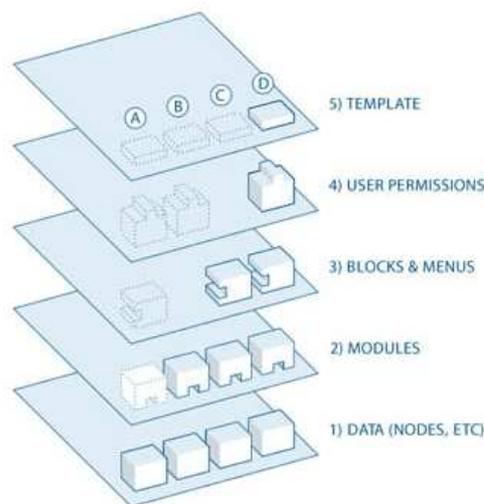


Ilustración 6: Arquitectura del CMS Drupal

CAPÍTULO IV: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Con el fin de seguir la política establecidas por el CMS y para lograr un mejor funcionamiento de los servicios de solicitud, disseminación selectiva y gestión de información, se decide usar la arquitectura modular que posee Drupal para la implementación de estos.

4.3 Diseño.

El diseño es la primera etapa técnica del proceso de desarrollo de un producto y consiste, fundamentalmente, en modelar el software que se va a desarrollar.

4.3.1 Diagramas de Paquetes.

Los diagramas de paquetes se usan para mostrar la organización de los paquetes y las interacciones entre estos.

En el desarrollo de los servicios a implementar, se agruparon los paquetes de acuerdo con los definidos en la Intranet del Polo Petrosoft, para organizar las actividades sobre un mismo contenido. Además, se añadieron 2 nuevos paquetes y se modificaron 2 de los existentes.

El diagrama propuesto incluye 7 paquetes (4 que propone el diseño de la Intranet de Petrosoft, 2 que se añaden y el del CMS utilizado).

A continuación se muestra cómo fue que quedaron conformados los paquetes de los servicios a desarrollar.

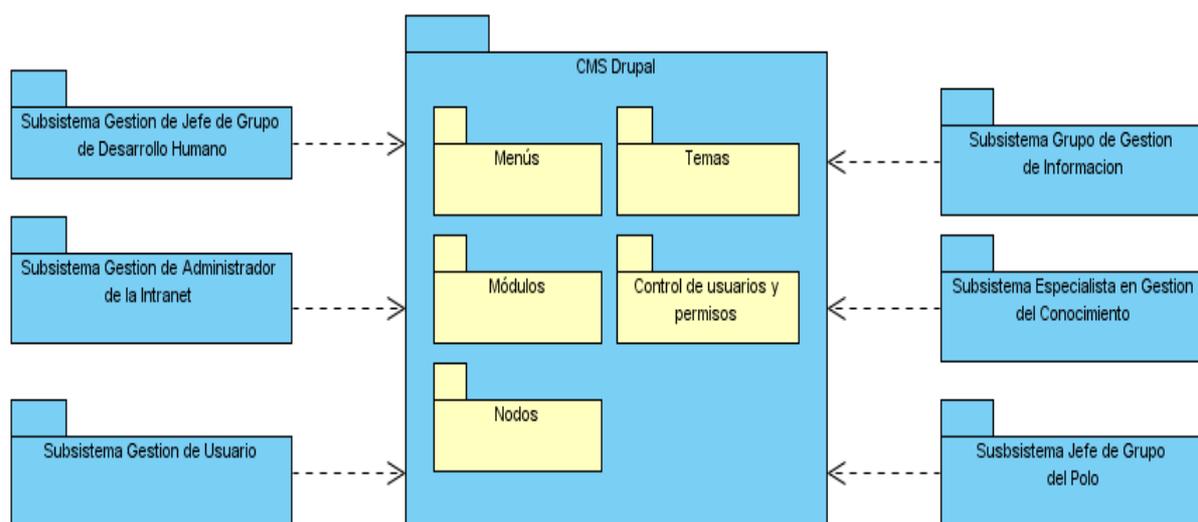


Ilustración 7: Diagrama de Paquetes del Sistema.

A continuación se mostrarán los paquetes modificados y las funcionalidades que fueron añadidas en estos paquetes, así como los paquetes añadidos (Subsistema Jefe de Grupo del Polo y Subsistema Especialista en Gestión del Conocimiento).

4.3.1.1 Subsistema Gestión de Usuarios.

En este paquete se añadieron nuevas funcionalidades realizadas por los usuarios.

- Visualizar Publicación.
- Subir trabajos Científicos a la aplicación.
- Visualizar Cronograma de presentación.

4.3.1.2 Subsistema Grupo de Gestión de la Información.

En este paquete se añadió una nueva funcionalidad realizada por el Grupo de Gestión de la Información.

- Gestionar Listado de Solicitudes de Información Científica.

4.3.1.3 Subsistema Jefe de Grupo del Polo

En este paquete se agruparon las principales funcionalidades realizadas por el Jefe de Grupo del Polo.

- Solicitar Información Científica.
- Gestionar Solicitudes de Información Científica.
- Recibir Información Selectiva.

4.3.1.4 Subsistema Especialista en Gestión del Conocimiento.

En este paquete se agruparon las principales funcionalidades realizadas por Especialista en Gestión del Conocimiento.

- Gestionar Listado de Solicitudes de Trabajos Científicos.
- Gestionar Cronograma de Presentaciones.
- Gestionar Evento.

4.3.2 Diagrama de clases del diseño.

Los diagramas de clases del diseño muestran una mejor visión del sistema y de la interacción entre las clases de este.

4.3.2.1 Diagrama de clases del diseño de Drupal.

Drupal contiene una única página servidor, la cual genera el contenido de la página final. Las páginas pueden o no contener formularios. Ello depende del módulo en cuestión y del propósito que tenga el mismo. (Vázquez 2009)

CAPÍTULO IV: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

A continuación se muestra el diagrama de clases del diseño de Drupal para tener una visión más clara de cómo es el funcionamiento de este CMS. Además, se muestran también los diagramas de diseño de los casos de uso solicitar información, gestionar solicitudes de información y gestionar listado de solicitudes de información. Los demás diagramas de los restantes casos de uso se pueden observar en la sección Anexos del presente documento. Estos diagramas se realizaron a partir del estudio realizado sobre el funcionamiento del CMS y de acuerdo a las características que deben presentar los servicios.

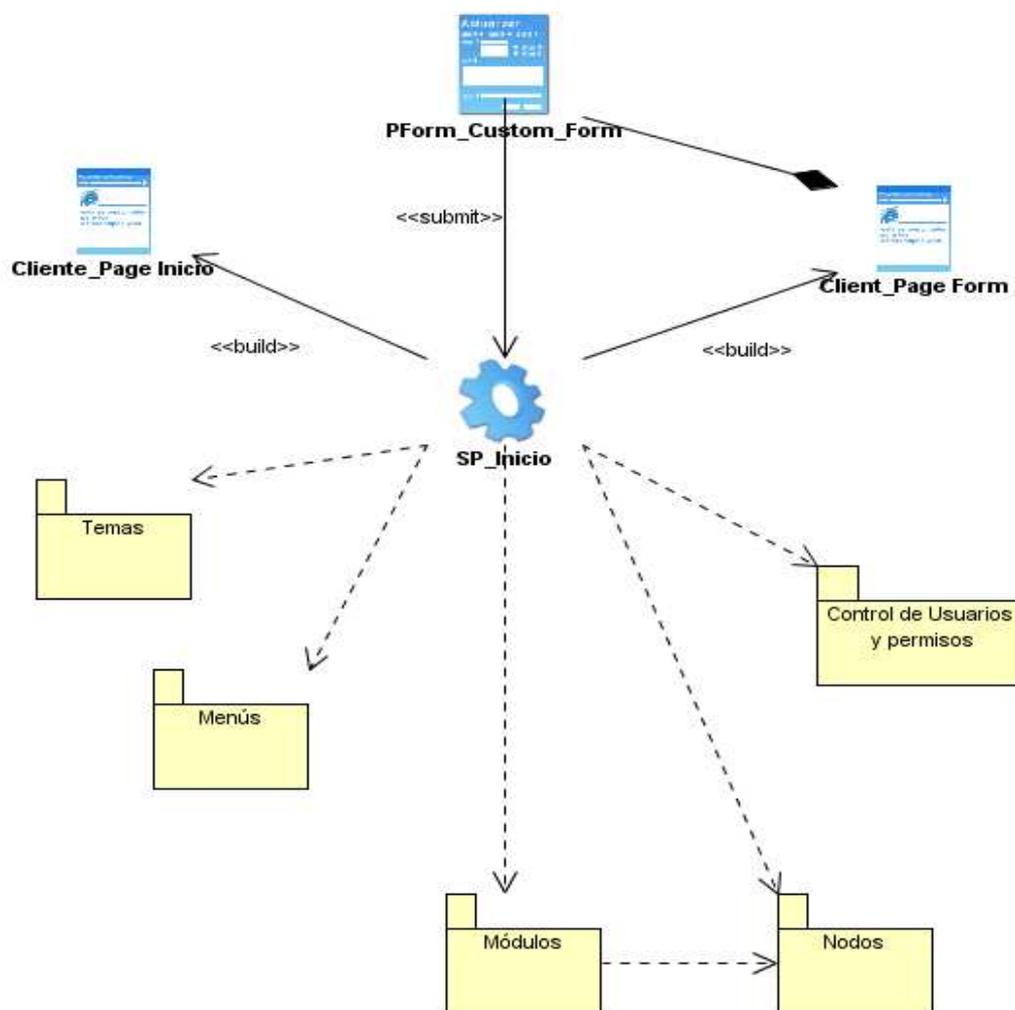


Ilustración 8: Diagrama de clases del diseño de Drupal.

4.3.2.2 Diagrama de clases del diseño del caso de uso Solicitar Información.

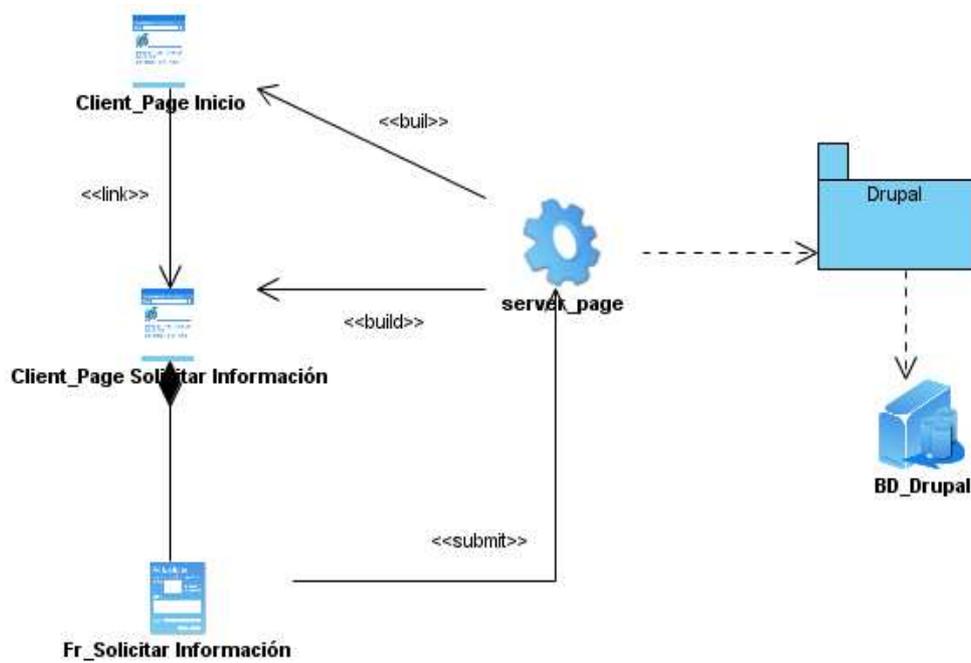


Ilustración 9: Diagrama de clases del diseño del caso de uso Solicitar Información.

4.3.2.3 Diagrama de clases del diseño del caso de uso Gestionar Solicitudes de Información.

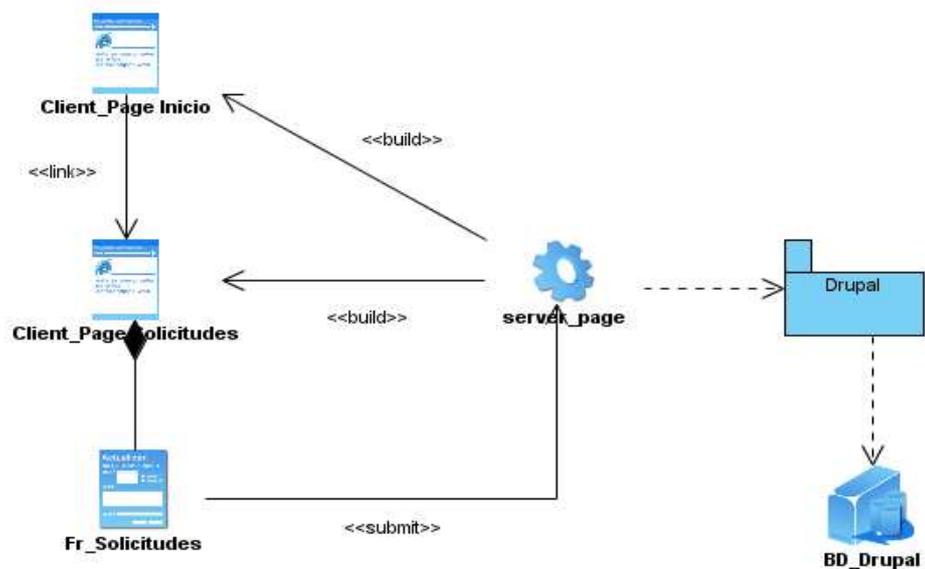


Ilustración 10: Diagrama de clases del diseño Gestionar Solicitudes de Información.

4.3.2.4 Diagrama de clases del diseño del caso de uso Gestionar Listado de Solicitudes de Información.

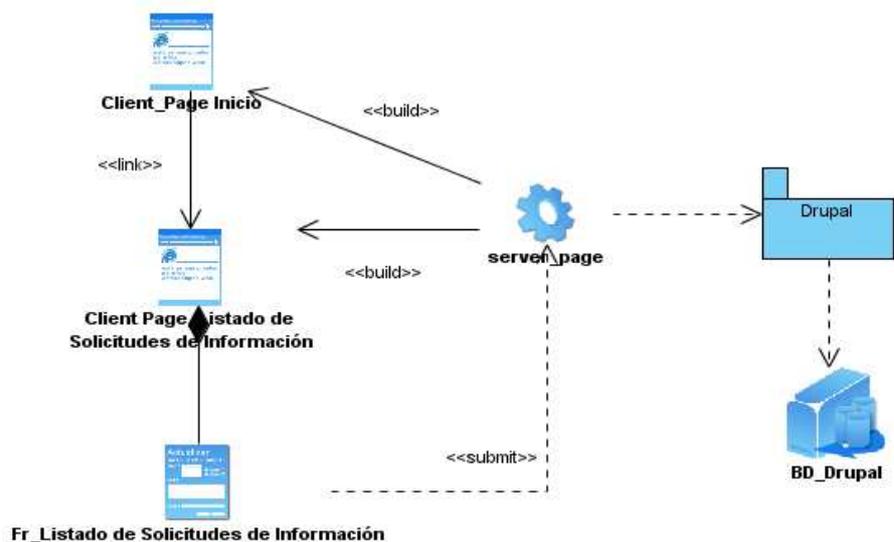


Ilustración 11: Diagrama de clases del diseño Gestionar Listado de Solicitudes de Información.

4.3.2.5 Diagrama de clases del diseño del caso de uso Gestionar Información.

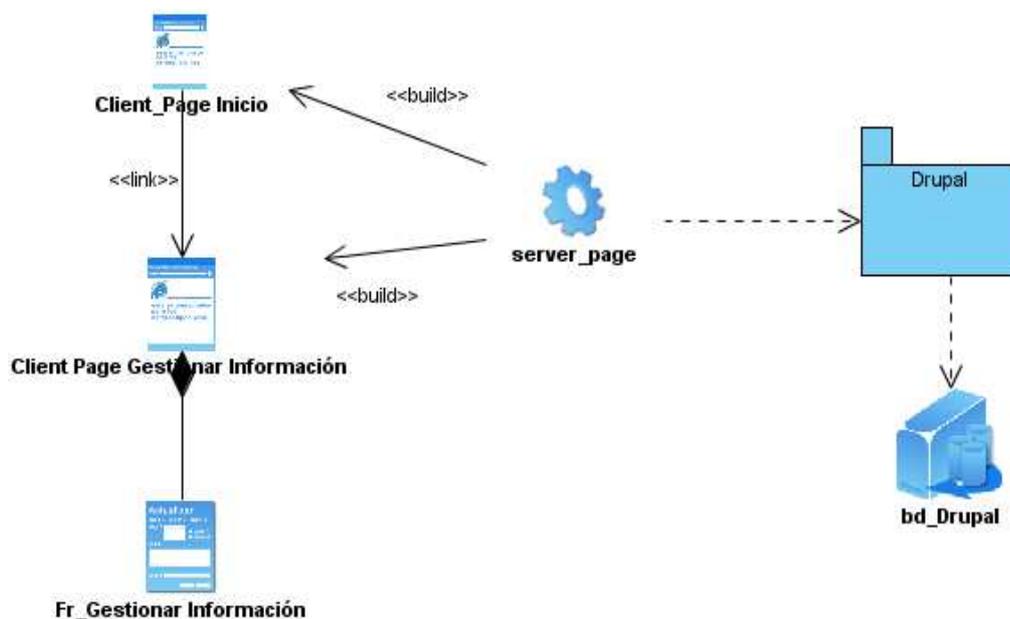


Ilustración 12: Diagrama de clases del diseño del caso de uso Gestionar Información.

4.3.3 Patrones de Diseño.

Los patrones de diseño consisten en una descripción de clases y objetos comunicándose entre sí, adaptada para resolver un problema de diseño general en un contexto particular. De manera general un patrón de diseño identifica clases, instancias, roles, colaboraciones y la distribución de responsabilidades. El uso de estos patrones ha contribuido a fomentar la flexibilidad y extensibilidad en los diseños y han demostrado ser una forma muy útil y exitosa de reutilizar diseños. (Vázquez 2009).

Estos patrones facilitan el aprendizaje de los diseñadores compendiando conocimientos ya existentes, estandarizan el modo de realizar el diseño, proporcionan elementos reusables en el diseño de software y establecen un vocabulario común entre los diseñadores.

Los principales patrones de diseño son los conocidos como GoF (del inglés Gang Of Four, en español grupo de los cuatro). En este grupo de patrones se encuentran los:

- Patrones creacionales (Abstract Factory, Builder, Factory Method, Prototype, Singleton).
- Patrones estructurales (Adapter, Bridge, Decorator, Façade, entre otros).
- Patrones de comportamiento (Command, Observer, State, Visitor, entre otros).
- Patrones de interacción (se utilizan para posibles soluciones de las interfaces de una web).

También están los patrones de diseño orientado a objeto para la asignación de responsabilidades o comúnmente denominados patrones GRASP (de inglés General Responsibility Assignment Software Patterns), que como su nombre lo indica son los encargados de realizar responsabilidades según la función de cada uno de estos.

Los patrones de diseño GRASP son los siguientes:

- Experto en Información: Asigna responsabilidades a los objetos.
- Creador: Identifica quien debe ser el responsable de la creación o instancias de clases u objetos.
- Alta Cohesión: Sugiere que la información almacenada en una clase sea la relacionada con esta.
- Bajo Acoplamiento: Se utiliza para mantener las clases lo menos relacionadas posibles para en caso de que ocurran cambios, estos repercutan lo menos posibles en las otras.
- Controlador: Controla las acciones del sistema, o sea, cuando se realiza una determinada acción en el sistema la envía al encargado de realizarla.

A continuación se hace referencia a los patrones que usa el CMS Drupal y los que serán usados en la implementación de los servicios a desarrollar.

4.3.3.1 Patrones de Diseño del CMS Drupal.

Drupal usa varios patrones de diseño. Entre estos se encuentran:

- Singleton: Garantiza que una clase tenga solamente una instancia y proporciona un punto de acceso global a ella. Si se piensa en los módulos y temas de Drupal como objetos, entonces se puede decir que este sigue el patrón Singleton, ya que estos objetos no encapsulan datos.
- Decorador: Este patrón añade dinámicamente funcionalidades a un objeto. Esto permite no tener que crear sucesivas clases que hereden de la primera incorporando la nueva funcionalidad, sino otras que la implementan y se asocian a la primera. Drupal hace un uso extensivo de este patrón. El polimorfismo con el objeto nodo es un ejemplo claro, pero esto es solo una parte de la potencialidad del sistema de nodos de Drupal.
- Observer: Este patrón define una dependencia del tipo uno-a-muchos entre objetos y cuando un objeto cambia su estado este se encarga de notificar el cambio a los objetos dependientes de él. Drupal aplica las modificaciones al vocabulario del sistema de taxonomías a todos los módulos que la implementan.

4.3.3.2 Patrones de diseño que se emplearán en la implementación de los servicios a automatizar.

Para dar solución a la implementación de los casos de uso planteados para el desarrollo de los servicios de gestión de información se aplican los patrones de diseño que usa el CMS Drupal y además, los patrones GRASP de asignación de responsabilidades experto, creador, alta cohesión y bajo acoplamiento.

En las capas de lógica del negocio se aplicó el principio de asignación de responsabilidades, o sea, cada clase implementa las funcionalidades de acuerdo con la información que posee. Ejemplo de esto es la clase solicitudes, encargada de gestionar toda la información relacionada con las solicitudes de información realizadas por los jefes de grupo del polo.

El patrón creador permitió identificar las clases responsables de crear objetos o instancias de otras clases. Para evidenciarlo se puede apreciar en las clases existentes, cuando se implementa el método `function_construct ()`, que es el encargado de crear objetos de tipo acceso a datos. Este método se utiliza para acceder a la clase acceso a datos.

Los patrones alta cohesión y bajo acoplamiento están estrechamente relacionados, ya que, a un mayor grado de cohesión un menor grado de acoplamiento. Esto se pone de manifiesto en todas las clases de la lógica del negocio, ya que cada clase almacena la información relacionada con esta y se trata de que estas estén lo menos relacionadas posible con las demás.

4.3.4 Diseño de la Base de Datos.

Las bases de datos no son más que son colecciones almacenadas de información de un sistema. Estas poseen características peculiares, y por tanto es necesario realizar un diseño adecuado de las mismas.

Las clases persistentes son aquellas clases de la base de datos cuya información va a permanecer, independientemente del tiempo y el espacio. Para diseñar la base de datos se parte del diagrama de clases persistentes y seguidamente se da paso a establecer las relaciones que se evidencien entre estas.

Los servicios que se implementaron están desarrollados con Drupal, y para el almacenamiento de la información de los mismos se utilizó la base de datos de este CMS, agregándose 2 tablas nuevas (solicitudes y evento) y modificándose la tabla node_dir_listing_content. Por tal motivo el diagrama de clases persistentes no es muy complejo.

A continuación se muestra como quedó conformado.

4.3.4.1 Diagrama de clases persistentes del sistema.

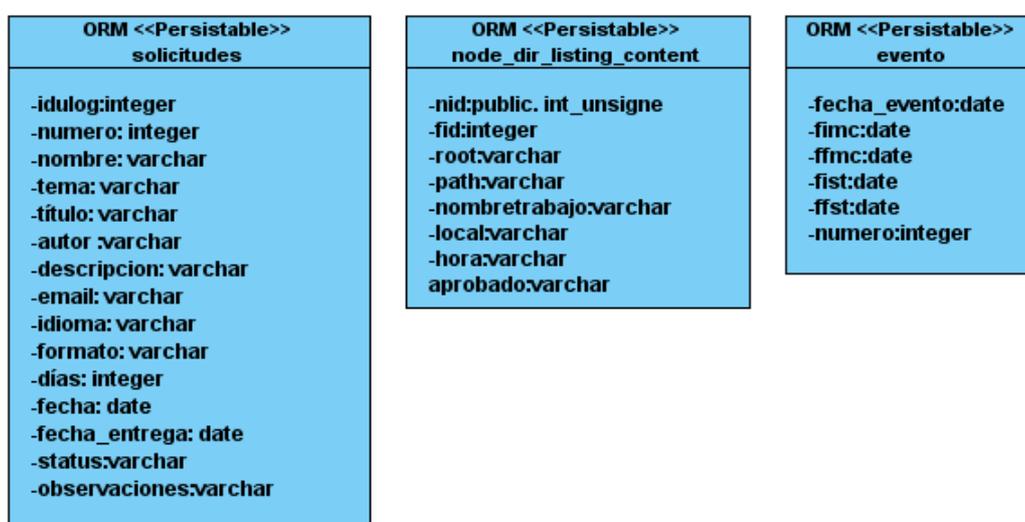


Ilustración 13: Diagrama de Clases Persistentes.

4.3.4.2 Modelo de datos del sistema.

A partir del diagrama de clases persistentes se obtuvo el correspondiente modelo de datos. En el presente y teniendo en cuenta la explicación anterior, solo se ilustraron las tablas que se modificaron y añadieron.

solicitudes			node_dir_listing_content			evento		
nombre	varchar(60)	Nullable = true	nid	int4	Nullable = false	fecha_evento	date	Nullable = true
tema	varchar(80)	Nullable = true	+fid	int4	Nullable = false	fimc	date	Nullable = true
titulo	varchar(80)	Nullable = true	root	varchar(255)	Nullable = false	ffmc	date	Nullable = true
autor	varchar(80)	Nullable = true	path	varchar(255)	Nullable = false	fist	date	Nullable = true
email	varchar(60)	Nullable = true	nombretrabajo	varchar(80)	Nullable = true	ffst	date	Nullable = true
descripcion	varchar(1000)	Nullable = true	local	varchar(2147483647)	Nullable = true	numero	int4	Nullable = true
idioma	varchar(60)	Nullable = true	hora	varchar(20)	Nullable = true			
fecha	date	Nullable = true	aprobado	varchar(15)	Nullable = true			
fecha_entrega	date	Nullable = true						
observaciones	varchar(1000)	Nullable = true						
status	varchar(20)	Nullable = true						
dias	int4	Nullable = true						
idulog	int4	Nullable = true						
numero	int4	Nullable = false						

Ilustración 14: Modelo de Datos del Sistema.

4.3.4.3 Descripción de las Tablas.

A continuación se hace una descripción de las tablas que fueron añadidas a la base de datos, y de la tabla que fue modificada (node_dir_linting_content) para un mejor entendimiento de las mismas.

4.3.4.3.1 Descripción de la Tabla Solicitudes.

Nombre: Solicitudes.		
Descripción: Esta tabla es la encargada de almacenar la información relacionada con las solicitudes de información que se realicen en el sistema.		
Atributo	Tipo	Descripción
Idulog	integer	Almacena el identificador del usuario que se registra en el sistema.
Número	integer	Almacena el número de la solicitud.

CAPÍTULO IV: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Nombre	varchar	Almacena el nombre de la persona que realiza la solicitud.
Tema	varchar	Almacena el tipo de información que se desea solicitar.
Título	varchar	Almacena el título del artículo.
Autor	varchar	Almacena el autor del artículo.
Descripción	varchar	Almacena la descripción de la información que el solicitante necesita.
Email	varchar	Almacena el correo electrónico del solicitante.
Idioma	varchar	Almacena el idioma en que quiere recibir la información.
Formato	varchar	Almacena el formato en que desea recibir la información.
Días	integer	Almacena la frecuencia de días con que desea recibir la información.
Fecha	date	Almacena la Fecha actual del día en que se realiza la solicitud.
Fecha_entrega	date	Almacena la fecha en que el solicitante desea que se le entregue la información.
status	varchar	Almacena el estado en que se encuentra la solicitud.
Observaciones	varchar	Almacena el comentario que el solicitante puede realizar a cerca de la solicitud que efectuó.

Tabla 20: Descripción de la Tabla Solicitudes.

4.3.4.3.2 Descripción de la Tabla node_dir_listing_content.

Nombre: node_dir_listing_content.		
Descripción: Esta tabla es la encargada de almacenar la información relacionada con los trabajos que se suben a la aplicación.		
Atributo	Tipo	Descripción

CAPÍTULO IV: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA

Nid	Int_unsigne	Almacena el id del usuario autenticado.
Fid	integer	Almacena el id del documento que se encuentra en el servidor.
Root	varchar	Almacena la dirección donde se encuentra el documento.
Path	varchar	Almacena el nombre con que se subió el documento a la aplicación.
Nombretrabajo	varchar	Almacena el nombre del trabajo que se subió a la aplicación.
Local	varchar	Almacena el local de presentación del trabajo.
Hora	varchar	Almacena la hora de presentación del trabajo.
Aprobado	varchar	Almacena el estado del trabajo después de revisado. Este estado puede ser aprobado o rechazado.

Tabla 21: Descripción de la Tabla node_dir_listing_content.

4.3.4.3.3 Descripción de la Tabla Evento.

Nombre: Evento.		
Descripción: Esta tabla es la encargada de almacenar la información relacionada la fecha del evento y con las opciones visuales de mostrar el cronograma y la opción subir trabajos a la aplicación.		
Atributo	Tipo	Descripción
Fecha_evento	date	Almacena la información relacionada con la fecha de realización del evento.
Fimc	date	Almacena la información relacionada con la fecha de inicio a partir de la cuál será mostrado el cronograma.
Ffmc	date	Almacena la información relacionada con la fecha de fin que será mostrado el cronograma.
Fist	date	Almacena la información relacionada con la fecha

		de inicio para subir trabajos a la aplicación.
Ffst	date	Almacena la información relacionada con la fecha de fin para subir trabajos a la aplicación.
Número	integer	Almacena el número del evento.

Tabla 22: Descripción de la Tabla Evento.

4.3.5 Diagrama de componentes del sistema.

Un diagrama de componentes muestra como se encuentra dividido un software y las relaciones entre los componentes. Los diagramas de componentes presentan un nivel de abstracción más elevado que los diagramas de clases, aunque los componentes usualmente se encuentran implementados por clases.

Se utilizan para modelar las diferentes vistas de un sistema (estática y dinámica) y para mostrar que componentes pueden ser utilizados en común por varios sistemas o entre las diferentes partes de este.

A continuación se muestran los diagramas de componentes de los casos de uso más críticos que se implementarán, los demás diagramas de componentes serán incorporados en la sección Anexos del presente trabajo.

4.3.5.1 Diagrama de componentes del caso de uso Solicitar Información Científica.

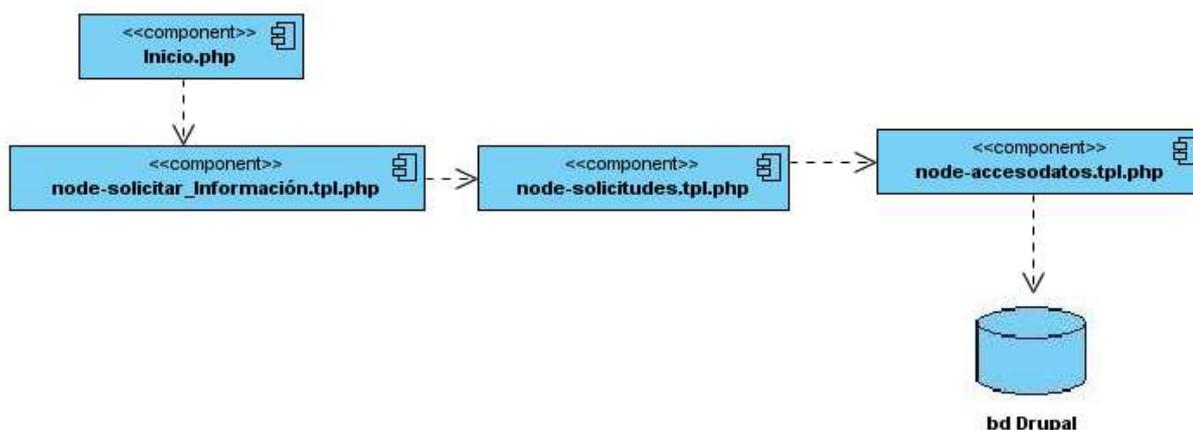


Ilustración 15: Diagrama de Componentes del Caso de Uso Solicitar Información Científica.

4.3.5.2 Diagrama de componentes del caso de uso Gestionar Solicitudes de Información Científica.

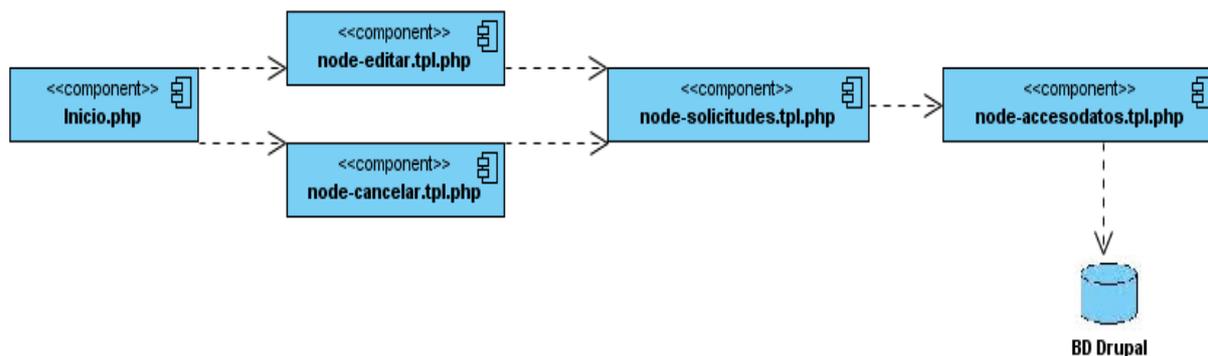


Ilustración 16: Diagrama de Componentes del Caso de Uso Gestionar Solicitudes de Información Científica.

4.3.5.3 Diagrama de componentes del caso de uso Gestionar Listado de Solicitudes de Información Científica.

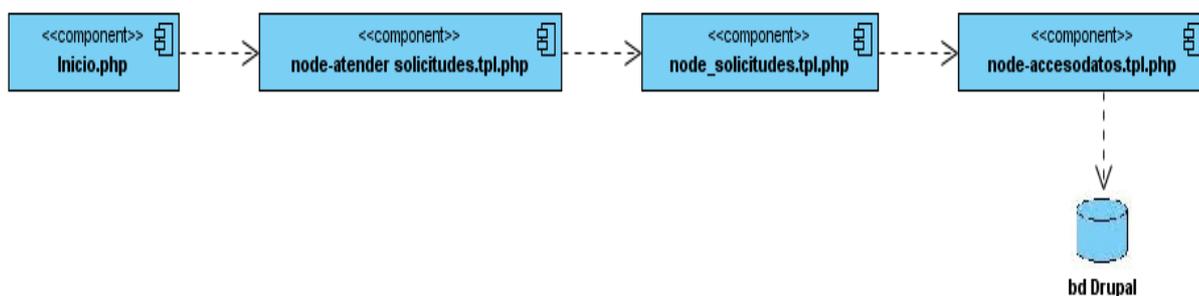


Ilustración 17: Diagrama de componentes del caso de uso Gestionar Listado de Solicitudes de Información Científica.

4.3.5.4 Diagrama de componentes del caso de uso Gestionar Información.

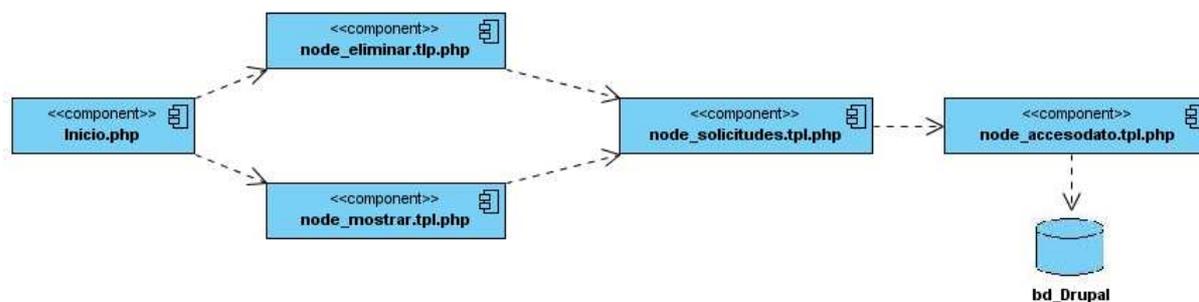


Ilustración 18: Diagrama de componentes del caso de uso Gestionar Información.

4.3.6 Diagrama de Despliegue del Sistema.

Un diagrama de despliegue es un grafo de nodos unidos por conexiones de comunicación. Un nodo puede contener instancias de componentes software, objetos, procesos, siendo este una unidad de computación de algún tipo (computadora, servidor).

A continuación se muestra el diagrama de despliegue de los servicios desarrollados, teniendo en cuenta el despliegue de la Intranet del Polo Petrosoft, pues los servicios implementados se incorporarán a esta.

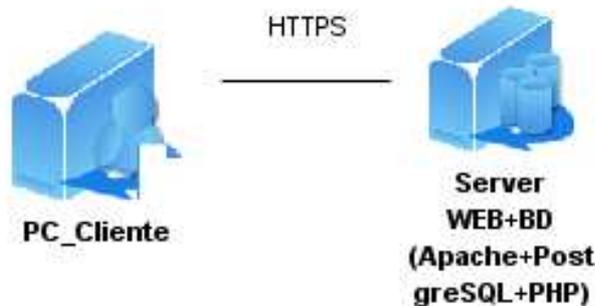


Ilustración 19: Diagrama de Despliegue del Sistema.

4.4 CONCLUSIONES.

En este capítulo se construyeron los artefactos necesarios para realizar una correcta implementación de los servicios de solicitud, diseminación selectiva y gestión de información. Se obtuvo el modelo de clases persistentes, y el modelo de datos de la base de datos; también se obtuvo el diagrama de componentes de los casos de uso del sistema para un mejor entendimiento y se mostró el despliegue de los servicios de gestión de información.

CAPÍTULO V: VALIDACIÓN DE LA PROPUESTA.

5.1 INTRODUCCIÓN.

Al culminar el desarrollo de los servicios para la solicitud, diseminación selectiva y gestión de la información, se procede a la validación de los mismos por parte de los expertos en el tema para un posterior empleo de estos por otros equipos de trabajo.

5.2 Análisis de la validación.

Los servicios desarrollados fueron validados por 8 expertos; 5 en gestión de información, cuya experiencia en el tema oscila entre 5-10 años y 3 personas graduadas de Ingenieros Informáticos en la Universidad cuya experiencia oscila entre 1-4 años.

Estas personas se consideraron expertos debido al desempeño laboral que poseen dentro de la Universidad de las Ciencias Informáticas y a los conocimientos para responder el cuestionario de validación.

Los expertos relacionados con la validación de las aplicaciones desempeñan los siguientes roles:

- **Experto 1:** Especialista de Procedimientos Técnicos de la Dirección de Información de la Universidad de las Ciencias Informáticas, Licenciado en Bibliotecología.
- **Experto 2 y 3:** Especialistas Generales en la Dirección de Información de la Universidad de las Ciencias Informáticas, Licenciado en Idioma y Licenciado en Filología respectivamente.
- **Experto 4:** Profesor de idioma inglés de la facultad 9, Licenciado en Idioma Extranjero.
- **Experto 5:** Especialista de Información en el centro Tecnomática, licenciado en Idioma Extranjero.
- **Experto 6, 7 y 8:** Desarrollador de Interfaz de Usuario de UCIFAR, profesor de Probabilidad y Estadística de la facultad 9 y Jefe de Desarrollo respectivamente, Ingenieros Informáticos.

Para evaluar la propuesta construida, se tuvo en cuenta los siguientes indicadores:

- Aspectos técnicos y estéticos.
- Aprendizaje.
- Diseño de la aplicación.
- Usabilidad de la aplicación.

CAPÍTULO V: VALIDACIÓN DE LA PROPUESTA

Estos indicadores se seleccionaron con el propósito de obtener un criterio específico a cerca de los puntos que se consideran más importantes en los servicios desarrollados, para, de este modo, conocer si fueron cumplidos los objetivos propuestos en la presente investigación.

El método de validación empleado fue el de encuesta, la cual fue entregada a los expertos y cada uno, de manera independiente emitió su criterio a cerca del trabajo desarrollado.

Los mismos accedieron a la aplicación y realizaron las acciones pertinentes que les permitirían luego emitir una opinión.

5.3. Elaboración de la encuesta.

La encuesta aplicada a los expertos contaba con 4 indicadores a evaluar, y estos por su parte, estaban conformados por un sub conjunto de indicadores.

La calificación de los sub indicadores era sobre la base de 5 puntos, aunque en algunos no había que calificar si no que responder. Las respuestas que se podían dar eran: Sí, Medio y No, donde a la hora de procesar los datos, las calificaciones numéricas que correspondían a cada una de las respuestas eran 5,3 y 2 respectivamente.

La calificación final de cada uno de los indicadores será el valor entero de la suma del criterio emitido por los expertos, para que sea más exacto el cálculo final del puntaje.

Los resultados arrojados en las encuestas realizadas a los expertos fueron los siguientes.

Indicadores Experto	1	2	3	4
1	19	15	20	25
2	17	13	20	25
3	19	15	20	25
4	20	15	16	20
5	18	15	18	25
6	19	13	20	25
7	18	13	18	25
8	17	15	16	25

Total x Indicadores	147	114	148	195
% que representa del total.	92%	95%	93%	98%

Tabla 23: Resultados de la encuesta realizada a los expertos.

5.4. Procesando los datos.

Una vez concluida la fase de procesamiento de los datos, arrojados en las encuestas realizadas a los expertos, se pudo apreciar que:

- El 14 % de los expertos coincide en que los servicios desarrollados presentan un 100 % de desarrollo en aspectos técnicos y estéticos, un 42 % cree que presentan solo un 95 % en este aspecto, el 29 % de los expertos piensa que los servicios desarrollados presentan un 90 % y el resto opina que solo presentan un 85%.
- En cuanto al aprendizaje que presentan los servicios, se puede señalar que el 63% de los expertos coinciden en estos cumplen 100% con este indicador, el resto de los expertos opina que solo cumplen con el 86%.
- En cuanto a diseño de los servicios desarrollados, el 50% de los expertos opina que cumplen 100% con esto, un 25% cree que cumplen con un 90% y el resto de los expertos que solo cumplen un 80%.
- El aspecto relacionado con la usabilidad arrojó que el 87% de los expertos concuerdan en que los servicios desarrollados son usables 100%, el resto creen que solo presentan un 80% de usabilidad.

En sentido general se pudo apreciar, en cuanto a aspectos establecidos que:

- Los servicios desarrollados cumplen en un 92% con los aspectos técnicos y estéticos que debe presentar.
- En cuanto al aprendizaje que debe presentar los servicios, estos cumplen en un 95% con lo que debe proporcionar.
- El diseño de los servicios desarrollados representa el 93% del total que debe poseer esta.
- La usabilidad que presentan los servicios es de un 98% del total que debía poseer.

Teniendo en cuenta los resultados arrojados se puede decir que los servicios desarrollados para la solicitud, diseminación selectiva y gestión de información, satisfacen en más del 80%

los parámetros establecidos para la evaluación de los mismos, por lo que se consideran aptos para ser usados por otras personas.

5.6 CONCLUSIONES.

En este capítulo se demostró por el método de los expertos que los servicios implementados están aptos para ser usados por otras personas y que su uso resultaría muy beneficioso en el desarrollo de las tareas realizadas en el Polo Petrosoft.

CONCLUSIONES GENERALES.

Al concluir la investigación se logró implementar los servicios para la gestión de la información en el Polo Productivo Petrosoft de la Facultad 9 de la Universidad de las Ciencias Informáticas, con lo cual se dió cumplimiento al objetivo general planteado en la misma.

De igual manera se puede decir que:

- Se realizó la modelación del negocio y se identificó detalladamente, cómo se realizaban en la actualidad los procesos de solicitud y envío de información, logrando una visión más clara del negocio y del funcionamiento de este.
- Se capturaron los requisitos funcionales y no funcionales que el sistema debía poseer y se realizó la modelación del sistema, lo que trajo consigo una mejor organización del trabajo a la hora de implementar los servicios de solicitud, diseminación selectiva y gestión de la información.
- Se emplearon patrones de diseño para un mejor desarrollo en la implementación de los servicios y, aprovechando las ventajas que presenta el CMS Drupal se utilizó la arquitectura que propone el mismo, logrando una mejor interacción de los servicios implementados con Drupal y un mejor funcionamiento de los mismos.

En sentido general se puede decir que los servicios implementados cumplieron con los requisitos propuestos en la investigación y los expertos que los evaluaron los calificaron de aptos para ser utilizados por otras personas.

RECOMENDACIONES.

Se recomienda para un posterior desarrollo los servicios:

- Actualizar la Intranet del Polo.
- Agregar más campos al formulario de solicitud de información, con el propósito de realizar búsquedas más específicas.
- Permitir que no solo se suban a la aplicación los resúmenes de los trabajos en formato de documentos Word, sino también en otros formatos, pues nos encontramos en un mundo muy cambiante, donde la tecnología avanza con mucha rapidez. Además, los documentos en formato Word son muy propensos a ser plagiados debido a que este formato presenta poca seguridad.
- Incluir una ayuda con el objetivo de viabilizar posibles dudas que puedan tener usuarios comunes respecto a las funcionalidades de los servicios desarrollados.
- Enviar los correos de aceptación o rechazo de trabajos científicos desde la aplicación y no de forma manual.

TRABAJOS CITADOS.

1. Amaro Calderón, S. D., Valverde Rebaza. Jorge Carlos (2007). Metodologías Ágiles. Trujillo-Perú, Universidad Natural de Trujillo.
2. Angel, M. (2010). Archivos para Servidores de Bases de Datos. 2010.
3. Bartle, P. (2009). "Información para la gestión y Gestión de la información." Retrieved 10,12, 2009, from <http://www.scn.org/mpfc/modules/mon-miss.htm>.
4. Bustelo, C., Amarilla, Raquel. (2001). "GESTIÓN DEL CONOCIMIENTO Y GESTIÓN DE LA INFORMACIÓN."
5. Charles Lamb, J. H., CarlMcDaniel (2002). *Marketing* (International Thomson Editores.
6. Ciberaula. (2006). "Linux Apache Intro." Retrieved 28, 01, 2010, from http://linux.ciberaula.com/articulo/linux_apache_intro.
7. Dante, G. P. (2004). Gestión de la Información: Dimensiones e Implementación para el éxito organizacional. Argentina, Nuevo Pharadigma.
8. Española, R. A. (2009). Diccionario de la Real Ademia Española.
9. Ferrer, D. (2007). "¿Qué es Joomla?" Retrieved 27,01, 2010, from <http://www.maestrosdelweb.com/editorial/que-es-joomla/>.
10. Fowler, M. (2003). "La Nueva Metodología." Retrieved 11,01, 2010, from <http://www.programacionextrema.org/articulos/newMethodology.es.html>
11. González, D. C. C. Y. C. (2009). Portal de Software Libre de Cuba. Propuesta para su desarrollo. Cuba, Universidad de las Ciencias Informáticas.
12. Hernández, A. O. (2009). Análisis y Diseño del Sistema de Producción para la Empresa de Perforación y Extracción de Petróleo de Occidente. . Ciudad de la Habana, Universidad de las Ciencias Informáticas.
13. Hernandis, J. A. (2005). "Visual Paradigm for UML." Retrieved 18, 01, 2010, from <http://www.versionzero.com/noticia/210/visual-paradigm-for-uml>.
14. Kotler Philip, B. P., Hayes Thomas (2004). El marketing de Servicios Profesionales Editorial Paidós SAICF.
15. Lévenez, É. (2007). "Computer Languajes History." Retrieved 20, 01, 2010, from <http://www.levenez.com.lang/> .
16. Mononeurona. (2007). "¿PHP, Python, ASP, Perl o JSP?" Retrieved 25, 01, 2010, from <http://www.mononeurona.org/pages/display/150> .
17. Murray, P. (2002). *Gestión-Información-Conocimiento*. 14.
18. Paradigm, V. (2007). "Visual Paradigm for UML." Retrieved 15, 01, 2010, from <http://www.visual-paradigm.com/> .
19. PostgreSQL. (1996). "PostgreSQL ", from <http://www.postgresql.org/>.
20. Pressman, R. S. (2001). Ingeniería del Software, un enfoque práctico
21. Rational. (2007). "Rose data Modeler." Retrieved 20, 01, 2010, from <http://www.rational.com.ar/herramientas/rosedatamodeler.html>.

22. Reyero, J. A. (2006). "Sobre Drupal." Retrieved 26,01, 2010, from <http://drupal.org.es/drupal>.
23. Rodríguez Luis, Z. G. E. R. (2009). Intranet Petrosoft. Ciudad de la Habana, Universidad de las Ciencias Informáticas.
24. Rodríguez Quintana, J. (2006). "Sistemas Distribuidos con Componentes." Retrieved 15,02, 2010, from <http://www.uv.mx/iiesca/revista4/distribuidos.html>.
25. Romero Heber, R. Y. (2009). *Gestión del conocimiento y la información en el Polo de Gestión Universitaria*. Ciudad de la Habana, Universidad de las Ciencias Informáticas.
26. Saavedra, G. R. (2009). *Gestión, Calidad y Agregación de valor en información*. Chile, Universidad de Chile.
27. Saint-Paul, A. (2008). Retrieved 1, 03, 2010, from http://www.tendencias21.net/La-informacion-es-fuente-de-poder-para-el-individuo_a2038.html.
28. Sánchez, M. A. M. (2004). "Metodologías de Desarrollo de Software." Retrieved 11, 1, 2010, from http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html.
29. Sandhusen, R. (2002). *Mercadotecnia*, Compañía Editorial Continental.
30. Stanton William, E. M., Walker Bruce, Mc Graw Hil (2004). *Fundamentos de Marketing*.
31. Thompson, I. (2006). "Definición de Servicios." Retrieved 9, 12, 2009, from <http://www.promonegocios.net/mercadotecnia-servicios/definicion-servicios.html>.
32. Vallado, O. G. (2008). *Diseminación Selectiva de Información en Bibliotecas Universitarias*. México, Universidad Nacional Autónoma de México.
33. Vázquez, D. A. H. E. M. (2009). *Sistema de Gestión para la Planificación Docente en la Facultad 5*. Cuba, Universidad de las Ciencias Informáticas.
34. Velezmoro, K. (2008). *Procesos Unificados y AUP. Gestión de Proyectos*. Perú. 2010.
35. Woodman, L. (1985). *Information management in large organizations* London ASLIB.

BIBLIOGRAFÍA.

1. Arquitectura Aplicaciones Web. [Online] www.MiTecnologico.com. [Cited: 04 23, 2010.] <http://www.mitecnologico.com/Main/ArquitecturaAplicacionesWeb>.
2. Barrero, Yeleny Serrano and Torres, Ekaterina Perdigón. 2010. Proceso de desarrollo de aplicaciones Web sobre la plataforma Drupal. Cuba : s.n., 2010.
3. Blasi, Emanuel. 2005. Resumen de patrones de Diseño. Argentina : s.n., 2005.
4. Comunic-Art. 2010. Tutorial de Drupal. [Online] 04 18, 2010. <http://www.cursosdrupal.com/content/arquitectura>.
5. Dominguez, Disnier Alberto Camejo. 2007. PROPUESTA DE LA ARQUITECTURA PARA EL SISTEMA INTEGRAL DEL. Cuba : s.n., 2007.
6. Drupal. [Online] [Cited: 04 26, 2010.] <http://drupal.org.es/drupal>.
7. Hernández, Rotney Rodríguez and Portales, Pablo Alberto Hernández. 2008. Propuesta de metodología Web basada en el CMS Drupal para el desarrollo del portal Web del CICPC. Cuba : s.n., 2008.
8. 2008. Kernel Error. [Online] Kernel Error, 2008. [Cited: 04 23, 2010.] <http://kernelerror.net/programacion/php/arquitectura-3-capas/>.
9. Larman, Craig. UML y Patrones. Canada : s.n.
10. SlideShare. 2010. El incomprendido Drupal. [Online] 04 15, 2010. <http://www.slideshare.net/develCuy/el-incomprendido-drupal>.
11. VanDyk, John K. 2008. Pro Drupal Development. Estados Unidos : Apress, 2008. ISBN-13 : 978-1-4302-0989-8.
12. Schmuller, Joseph. 2000. UML en 24 horas. s.l. : Prentice Hall, 2000. 968-444-463-X.

GLOSARIO DE TÉRMINOS.

AMA (American Marketing Association): Entidad de origen estadounidense que agrupa a los profesionales del marketing y periódicamente publica documentos relacionados con esto.

API (Interfaz de Programación de Aplicaciones): Conjunto de procedimientos usados por otro software como una capa de abstracción.

ASP (Active Server Page): Tecnología de tipo lado del servidor para páginas web generadas dinámicamente.

AUP (RUP Ágil): Metodología ágil de desarrollo de software.

BSD (Berkeley Software Distribution): Licencia de software libre otorgada principalmente para los sistemas BSD y que permite el uso del código fuente en software no libre.

CMS: Sistema de gestión de contenido (en inglés Content Management System), se utiliza para la creación y administración de contenido, actualmente se usa para denominar a los sistemas que sirven para administrar el contenido de páginas Web.

EGC (Especialista en Gestión del Conocimiento): Persona encargada de revisar los trabajos en el Polo Petrosoft.

EJB (Enterprise Java Bean): Modelo de componentes del lado del servidor.

GPL (General Public License): Licencia creada para proteger la libre distribución, modificación y uso del software.

Herramientas CASE (Ingeniería de Software Asistida por Ordenador): Herramientas informáticas destinadas a mejorar la productividad del software.

HTTP (HyperText Transfer Protocol): Protocolo usado para la transmisión de los datos a través de la red.

INTERNET: Conjunto de redes de ordenadores y equipos físicos conectadas entre sí desde diferentes partes del mundo.

JGP (Jefe de Grupo del Polo): Persona encargada de dirigir a un grupo de personas según una temática en el Polo Petrosoft.

JSP (Java Server Page): Tecnología que permite crear contenido dinámico web en forma de documento HTML, XML u otro tipo.

MSF (Microsoft Solution Framework): Metodología pesada de desarrollo de software.

Patrón: Es una solución a un problema de diseño no trivial que es efectiva y reusable. **PHP (Hypertext Pre-processor):** Lenguaje de programación del lado del servidor diseñado para la creación de páginas web dinámicas.

RAE (Real Academia Española): Institución española especializada en lexicografía, gramática, ortografía y bases de datos. Produjo el diccionario de la Real Academia.

RUP (Rational Unified Process): Es una estructura de desarrollo de software que suministra conocimientos recomendados y probados y una arquitectura configurable.

SGBD (Sistema de Gestión de Bases de Datos): Programas que permiten almacenar y acceder a la información de manera rápida. Son la interfaz entre la base de datos y el usuario.

SUN (Stanford University Network): Empresa informática productora de hardware y software.

UCI (Universidad de las Ciencias Informáticas): Universidad que se encarga entre otras cosas de desarrollar software.

UML (Lenguaje de Modelado Unificado): Lenguaje de modelado para visualizar, especificar y construir y documentar un sistema.

UNESCO (Organización de las Naciones Unidas para la educación, la ciencia y la cultura): Organismo especializado de las Naciones Unidas con el objetivo de contribuir a la paz y a la seguridad en el mundo mediante la educación, la ciencia, la cultura y las comunicaciones.

VP (Visual Paradigm): Herramienta de modelado profesional de procesos del desarrollo de software.

XP (Extreme Programming): Metodología ágil de desarrollo de software.

HARDWARE: Partes tangibles de una computadora.

SOFTWARE: Equipamiento o soporte lógico de una computadora. Conjunto de componentes lógicos necesarios para hacer posible la realización de una tarea específica.