



Universidad de las Ciencias Informáticas

DIRECCIÓN DE FORMACIÓN POSTGRADUADA

**Tesis presentada en opción al título de
Máster en Informática Aplicada**

“Propuesta para disminuir el tiempo de desarrollo en aplicaciones informáticas que gestionen información poco variable en el tiempo.”

AUTOR: Ing. Annia Arencibia Morales

TUTOR: MSc. Iván Pérez Mallea

La Habana, noviembre de 2012

“Año 54 de la Revolución”

DECLARACIÓN JURADA DE AUTORÍA Y AGRADECIMIENTOS

Yo *Annia Arencibia Morales* con carnet de identidad *84101317312*, declaro que soy la autora principal del resultado que expongo en la presente memoria titulada *Propuesta de disminución del tiempo de desarrollo de las aplicaciones informáticas que gestionen información poco variable en el tiempo*, para optar por el título de Máster en Informática Aplicada.

Este trabajo fue desarrollado durante los años 2010 - 2011 en colaboración con mis colegas de equipo *Ing. Rolando Pompa González, Ing. Dannier Flores Ramos, Ing. José Mojena Alpizar, Ing. Renán Vázquez Moreno y la Lic. Meybel Rivero Rivadeneira*, quienes me reconocen la autoría principal del resultado expuesto en esta memoria.

Finalmente declaro que todo lo anteriormente expuesto se ajusta a la verdad, y asumo la responsabilidad moral y jurídica que se derive de este juramento profesional.

Y para que así conste, firmo la presente declaración jurada de autoría en La Habana a los _ días del mes de _____ del año 2012.

Ing. Annia Arencibia Morales
Autor

RESUMEN

La investigación está encaminada a proponer una aplicación, que le permita a los programadores disminuir el tiempo de implementación de los sistemas informáticos, obviando el desarrollo de los nomencladores y centrando toda su fuerza de trabajo en la información central de las aplicaciones.

Para dar solución al problema planteado, se crea un sistema configurable y flexible ante posibles cambios de la información poco variable en el tiempo, Sistema Nomenclador, permitiendo el almacenamiento y la gestión de este tipo de información; sirviendo como mecanismo de apoyo a las aplicaciones que hacen uso de esta. La integración entre sistemas se logra a través de servicios Web XML, debidamente descritos, utilizando el Lenguaje de Descripción de Servicios Web (WSDL). El Sistema Nomenclador fue utilizado e integrado a un conjunto de sistemas, con dicha integración se disminuyó el tiempo y el costo de desarrollo, obteniéndose un sistema completamente configurable a las necesidades de los sistemas informáticos.

La integración es un hecho inevitable entre sistemas informáticos, pero debe estar debidamente identificada desde el instante en que se diseña el sistema; con el objetivo de que los desarrolladores centren su fuerza de trabajo solamente en la integración y desarrollo de los requisitos propios del negocio.

Palabras claves: *nomencladores, servicios Web XML, tiempo de desarrollo*

ÍNDICE

INTRODUCCIÓN	6
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	11
1.1 Conceptos asociados a la investigación	11
1.2 Situación problemática y objeto de automatización	14
1.3 Ambiente de desarrollo propuesto.....	17
1.3.1 Ontología.....	17
1.3.2 Modelo Vista Controlador (MVC)	18
1.3.3 XML	18
1.3.4 SOAP	19
1.3.5 WSDL.....	19
1.3.6 Servicios Web XML	20
1.3.7 Seguridad en aplicaciones web	21
1.4 Sistemas automatizados existentes asociados al campo de acción	22
1.4.1 Características de algunos sistemas internacionales.....	23
1.4.1.1 Sistema para Prestadores.....	23
1.4.1.2 Nomenclador de procedimientos y servicios.....	24
1.4.1.3 Nomenclador cartográfico para personas con deficiencia visual.....	25
1.4.2 Características de algunos sistemas nacionales.....	26
1.4.2.1 Registro de la Clasificación Internacional de Enfermedades (RCIE)	26
1.4.2.2 Registro de Problemas de Salud de la Atención Primaria (RPSAP)	26
1.5 Conclusiones parciales	27
Capítulo 2: PROPUESTA DE SOLUCIÓN	29
2.1 Descripción de la propuesta de solución	29
2.1.1 Especificación de requerimientos de software	32
2.1.2 Descripción de la arquitectura.....	33
2.2 Implementación de la propuesta de solución	35
2.3 Definición de los servicios web implementados	39
2.3.1 Información de un nomenclador dado su ID en el sistema.....	40
2.3.2 Información del nomenclador dado un nombre.....	40

2.3.3 Lista de hijos dado el identificador del padre	41
2.3.4 Visión integrada de los resultados de implementación.....	42
2.4 Conclusiones parciales	44
CAPÍTULO 3: ANÁLISIS Y VALORACIÓN DE LOS RESULTADOS	45
3.1 Validación de los resultados de integración	45
3.2 Encuesta aplicada a desarrolladores y especialistas participantes en la integración de la solución ...	48
3.3 Aplicación de la técnica IADOV para medir satisfacción	51
3.4 Análisis económico de la solución.....	53
3.5 Valoración de la innovación y aporte práctico de los resultados. Beneficios	54
3.6 Conclusiones parciales	56
CONCLUSIONES	57
RECOMENDACIONES.....	58
REFERENCIAS BIBLIOGRÁFICAS.....	59
ANEXOS	68

ÍNDICE DE FIGURAS Y TABLAS

Figura 1. Modelo ontológico	13
Figura 2. Estructura del nomenclador de especialidades médicas	16
Figura 3. Estructura del nomenclador de especialidades médicas después de realizado el cambio.....	16
Figura 4. Creación de nomencladores en DASUTeN	24
Figura 5. Registro de la Clasificación Internacional de Enfermedades (RCIE)	26
Figura 6. Registro de Problemas de Salud de la Atención Primaria (RPSAP)	27
Figura 7. Acceso a la información del sistema	30
Figura 8. Diagrama de actividad del proceso <i>Crear Nomenclador</i>	31
Figura 9. Estructura del framework Symfony.....	34
Figura 10. Diagrama de actividad del proceso Crear código	36
Figura 11. Visión integrada de los resultados de implementación	44
Figura 12. Representación gráfica de los requisitos funcionales por sistema	46
Figura 13. Representación gráfica de los tiempos de desarrollo teniendo en cuenta la integración	48
Figura 14. Grado de aceptación del Sistema Nomenclador.....	49
Figura 15. Aumento del desarrollo de las aplicaciones informáticas que gestionan nomencladores	50
Figura 16. Centralización de la información en un solo sistema informático	50
Figura 17. Desgaste físico del personal que desarrolla sistema informáticos que gestionan nomencladores	50
Tabla 1. Comportamiento de algunos indicadores de desempeño de los servicios web	20
Tabla 2. Requerimientos funcionales.....	33
Tabla 3. Grupos anatómicos principales.....	37
Tabla 4. Estructura del código ATC de la Metformina	37
Tabla 5. Estructura regular del código ATC de la Metformina	38
Tabla 6. Descripción de los parámetros de entrada y salida del método <i>GetNomencladorDadold</i>	40
Tabla 7. Descripción de los parámetros de entrada y salida del método <i>GetNomencladorLikeNombre</i>	40
Tabla 8. Descripción de los parámetros de entrada y salida del método <i>GetHijosDadoldPadre</i>	41
Tabla 9. Descripción de los parámetros de entrada y salida del método <i>CampoValor</i>	41
Tabla 10. Requisitos funcionales por sistemas	46

Tabla 11. Tiempo total de implementación por sistema	46
Tabla 12. Entorno de desarrollo de los sistemas	47
Tabla 13. Tiempo de integración de los sistemas expresado en meses	47
Tabla 14. Tiempo total de la implementación teniendo en cuenta la integración.....	47
Tabla 15. Cuadro lógico de V.A. IADOV	52
Tabla 16. Gasto de los sistemas antes de la integración y después de la integración	53
Tabla 17. Gasto total de desarrollo antes de la integración y después de la integración	54

INTRODUCCIÓN

Organizar la información ha sido una necesidad del hombre que ha trascendido con el paso de los años, este ha buscado formas de estructurarla permitiéndole un fácil acceso y actualización. Este cúmulo de datos relacionados entre sí, tiene forma y significado en todo momento, pero su valor está determinado a partir de que es poseído, concienciado y convertido en conocimiento por alguna entidad. (Delgado, 2006)

Un mismo conjunto de datos puede ser organizado de diversas formas y cada una de ellas dará cuenta de un modo particular de comprender la información: "...*mientras que la información puede ser infinita, las formas de estructurarla no lo son...*". Una vez que se tiene un lugar en el cual puede ser conectada la información, esta se vuelve mucho más útil. Para ello es necesario el conocimiento y análisis del tema, a partir de esta instancia reflexiva se podrán tomar decisiones en el campo de la organización y de la expresión gráfica que caracterizará a la propuesta. (Wurman, 2001)

Cada organización es pertinente para un tipo de información y su utilización puede ser una alternativa válida aunque no excluyente. En una misma información pueden utilizarse varios criterios en forma conjunta, es posible encontrar líneas de tiempo, ubicación geográfica, categorías y todo a su vez jerarquizado. Pero hay que tener en cuenta que existe un conjunto de información que no varía en un tiempo determinado (más conocida como nomencladores), que también puede ser categorizada en cualquiera de las formas antes mencionadas.

Nomenclador, según la Real Academia Española, es un Catálogo que tiene la nomenclatura de una ciencia y **Nomenclátor** es un catálogo de nombres, de pueblos, de sujetos y de voces técnicas de una ciencia o facultad (Diccionario, 2011). Teniendo en cuenta ambos conceptos se puede concluir que un nomenclador es una estructura de información organizada según la ciencia para la cual fue creado. La información organizada debe estar presente como un elemento básico en el diseño de las estrategias de la innovación tecnológica y en las acciones que de ella se deriven. La utilización de las tecnologías ha permitido compartir los recursos de información, quebrantar las fronteras geográficas y acelerar los procesos de comunicación, fomentando un nuevo paradigma, sin importar clases sociales o diferencia de

idioma. Disponer de una adecuada organización de la información ahorra tiempo y esfuerzo. (Delgado, 2006)

En la actualidad, la aplicación de las herramientas informáticas, requiere un sistema organizado que permita tener la capacidad de transferir la información de forma rápida y eficaz, logrando acortar los tiempos de desarrollo. Desde el proceso de conceptualización de cualquier sistema informático, se debe considerar la implementación de requerimientos que permitan el manejo de los nomencladores que el sistema necesitará y que no cambiarán de valor en un tiempo determinado. Una de las principales dificultades de las aplicaciones donde se hace uso de estos, la constituye la gestión de los mismos en el tiempo de desarrollo. Los desarrolladores pierden mucho tiempo y se alarga el fin de los proyectos por la demora en la implementación de los requerimientos: crear, obtener y actualizar dicha información.

Actualmente la mayoría de los sistemas desarrollados poseen la gestión de los nomencladores de forma aislada, implicando que todos la definan de manera similar, ajustando cada una al negocio al cual pertenece. En algunos casos presentan implementaciones muy básicas (crear, leer, actualizar y eliminar), lo cual atenta considerablemente contra las aplicaciones informáticas; si cambia esta información, el sistema puede correr el riesgo de tener que volver a implementarse desde cero. Existe un grupo de sistemas desarrollados o en fase de construcción, que han obviado la forma de organizar y gestionar los nomencladores, de manera que al cambiar la información nombrada, el sistema se ve afectado; por ello es importante considerar la existencia de requerimientos adecuados en el momento de desarrollarlos.

La implementación de los requerimientos que están relacionados con los nomencladores atrasa el desarrollo del sistema, en dependencia de la cantidad de nomencladores que necesite la información central. La información recogida en estos sistemas no puede ser usada por otras aplicaciones, a menos que cumplan con su estructura arquitectónica. Esto crea lagunas de información en todo el desarrollo informático, la misma información se puede ver representada de varias formas trayendo como consecuencia la descentralización de la información.

Una de las posibles formas de concebir sistemas de software en la actualidad, consiste en diseñar aplicaciones en las cuales puedan acoplarse componentes que pertenezcan a organizaciones diferentes para crear una aplicación más completa. Con esto se espera que, se reduzcan los costos de

interoperabilidad, mejore la productividad, aumente la flexibilidad en las aplicaciones, mejore la integración entre los sistemas y se reutilicen mejor los componentes de software. Considerando lo anterior se puede expresar que los servicios Web XML son una de las variantes por la que se ha apostado para garantizar la comunicación entre los componentes y soluciones desarrolladas.

Al detectar todos los problemas planteados anteriormente se formula el siguiente **Problema a resolver**: *¿Cómo disminuir el tiempo de desarrollo de las aplicaciones informáticas que gestionen información poco variable en el tiempo?*

Se plantea como **Objeto de estudio** el *proceso de gestión de la información poco variable en el tiempo durante el desarrollo de sistemas informáticos*. Definiéndose como **Campo de acción** el *proceso de gestión de la jerarquía de la información poco variable en el tiempo durante el desarrollo de sistemas informáticos*.

Se tiene como **Objetivo general**, *desarrollar un sistema informático, basado en un modelo ontológico, que gestione información poco variable en el tiempo y se integre a las aplicaciones informáticas durante su concepción, disminuyendo su tiempo de desarrollo*.

Objetivos específicos:

1. *Analizar el funcionamiento de los sistemas que gestionen información poco variable en el tiempo.*
2. *Diseñar y desarrollar un sistema informático que permita gestionar la información poco variable en el tiempo de los sistemas informáticos durante su desarrollo.*
3. *Validar los resultados verificando: la disminución del tiempo de desarrollo en las aplicaciones que se integren al sistema, reutilización y generalización de la solución obtenida.*

Para guiar la investigación y comprobar los resultados se propone la siguiente **Hipótesis**: *si se desarrolla un sistema informático basado en un modelo ontológico, que gestione información poco variable en el tiempo y se integra a las aplicaciones informáticas durante su concepción, se logrará disminuir el tiempo de desarrollo de estas.*

Entre los métodos utilizados en esta investigación se destacan los siguientes:

- **Métodos teóricos:**

- ✓ El método analítico-sintético: para descomponer el problema de investigación en dos procesos. Uno de ellos constituye la organización de la información, y el otro, la disminución del tiempo de desarrollo de las aplicaciones informáticas que gestionen nomencladores.
- ✓ El método inductivo-deductivo: para obtener un grupo de conocimientos, analizando los datos generales válidos para llegar a una conclusión particular, específicamente sobre la forma de organizar la información poco variable en el tiempo en un sistema informático.
- ✓ El método análisis histórico lógico: para analizar de forma analítica la trayectoria histórica real de los fenómenos, su evolución y desarrollo; estableciendo similitudes con la investigación, y obtener puntos en común entre lo desarrollado en la historia y la propuesta de solución. Además de identificar en el tiempo, sistemas que realicen actividades semejantes a la propuesta.
- ✓ El método hipotético-deductivo: para la definición de la hipótesis de la investigación y para proponer nuevas líneas de trabajo a partir de los resultados parciales obtenidos, los cuales serán verificados posteriormente mediante métodos empíricos.
- ✓ El método de la modelación: con el fin de crear abstracciones e interpretar la realidad.
- ✓ Las encuestas: para la obtención de información. Se basan en el planteamiento de preguntas escritas y organizadas, aplicadas a un conjunto de personas, donde las respuestas fueron seleccionadas de acuerdo a criterios específicos.

La novedad y el aporte práctico de la investigación se concretan en un sistema informático que gestione nomencladores, permitiendo que los sistemas que lo utilicen disminuyan su tiempo de desarrollo. El funcionamiento de la solución obtenida no está limitado a su plataforma de desarrollo, sino que es totalmente escalable a cualquier otro tipo de sistema, el mismo posibilitará una amplia posibilidad de reutilización y generalización: uno de los principios más deseados en el desarrollo de software actual.

Otro elemento importante a considerar es que los analistas solo deberán centrarse en aquellos requerimientos que son propios de su negocio e incorporar posteriormente el uso de los requerimientos descritos en la solución propuesta; lo cual disminuye el tiempo de desarrollo de las aplicaciones, al reutilizar software ya probado y especializado en los procesos antes descritos. La solución obtenida posee una amplia relevancia en el contexto sobre el cual se desarrolla, debido a que el país viene realizando importantes esfuerzos en pos del desarrollo de la informática. El desarrollo de estos procesos garantiza una homogénea y estandarizada forma de gestionar la información.

El desarrollo del presente documento se encuentra estructurado en tres capítulos, en el primero de ellos “FUNDAMENTACIÓN TEÓRICA”, se tiene como objetivo abordar los diferentes aspectos teóricos que serán la base conceptual para el desarrollo de la propuesta de solución. Seguidamente en el capítulo, “PROPUESTA DE SOLUCIÓN”, se presenta un marco conceptual asociado al campo de acción, identificando las funcionalidades, requerimientos deseados y el objeto de automatización. Se realiza una modelación detallada de la construcción de la estructura de la aplicación e implementación. Finalmente, el tercer y último capítulo “ANÁLISIS Y VALORACIÓN DE LOS RESULTADOS”, aborda una valoración crítica tanto cuantitativa como cualitativa de los resultados obtenidos a partir de un conjunto de pruebas realizadas a la solución.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

El presente capítulo tiene como objetivo abordar los diferentes elementos que brindan la base teórico conceptual para el desarrollo de la solución propuesta; valorándose de forma crítica las tendencias y tecnologías actuales, así como los antecedentes asociados al campo de acción. De este modo, se podrá realizar una correcta interpretación de la situación problemática y del problema a resolver.

1.1 Conceptos asociados a la investigación

Los sistemas que necesitan de información poco variable en el tiempo, para su funcionamiento, tienen la necesidad de contar con toda esta información antes de empezar a desarrollar, con el objetivo de poder definir la dependencia de esta con el resto de la información que forma el eje vertical del desarrollo. Darse cuenta de toda la documentación necesaria y lograr un modelo de desarrollo que le permita al desarrollador poder avanzar en la implementación del sistema se puede hacer muy engorroso, todo depende de la documentación dada al equipo de desarrollo y la visión del mismo. Algunos autores han escrito sobre cómo se debe organizar la información y algunos conceptos relacionados al mismo. En su libro *Angustia Informativa*, Wurman¹ propone cinco formas para organizar la información:

La **Ubicación** es la forma natural de elegir cuando se está tratando de examinar y se compara información que proviene de diversas fuentes o lugares. Si se estuviera examinando una industria se necesitaría saber cómo está distribuida por el mundo. Los doctores usan las diferentes ubicaciones en el cuerpo como unidades o bloques para estudiar medicina.

El **Alfabeto**. Esta forma se presta para organizar cuerpos de información extraordinariamente grandes, tales como palabras en un diccionario o apellidos en una guía telefónica. La organización de la información por orden alfabético funciona cuando la audiencia o los lectores abarcan un amplio espectro de la sociedad y no podrían comprender la clasificación hecha siguiendo otra forma.

¹Arquitecto relacionado con la arquitectura y el diseño, con el objetivo de hacer la información comprensible. Autor de numerosos libros y origen de la denominación 'Arquitecto de Información'. Destacan entre sus títulos más relevantes *Information Anxiety* e *Information Architects*. Actualmente es uno de los gurús de la Arquitectura de Información.

El **Tiempo** funciona mejor como principio organizador para los acontecimientos que suceden a lo largo de duraciones fijas. Funciona con las exhibiciones, los museos y las historias. El tiempo es un marco fácilmente comprensible a partir del cual se pueden observar cambios y hacer comparaciones.

La **Categoría** pertenece a la organización de las mercancías. Categoría puede significar diferentes modelos, diferentes tipos e incluso diferentes cuestiones a ser contestadas, tales como un folleto que esté dividido en preguntas sobre una compañía. Este modo se presta muy bien para organizar ítems de importancia similar.

La **Jerarquía** organiza los ítems por su magnitud, desde pequeños hasta grandes, desde menos costosos hasta más costosos, por orden de importancia, etc. Este es el modo que se debe usar cuando queremos asignar valor o un cierto peso a la información, o cuando queremos utilizarla para estudiar algo como una industria o compañía. (Wurman, 2001)

Los conceptos mencionados anteriormente son la base para cualquier sistema informático que maneje este tipo de organización de la información o para quien quiera organizar la información en cualquiera de las ramas o lugares, pero en términos de sistema se deben manejar algunos conceptos que permitan a los desarrolladores implementar los requisitos. Se debe modelar un diagrama que permita comprender qué es lo primero que se debe crear. Para obtener una mejor visión de los conceptos fundamentales, a continuación se define por el autor un modelo ontológico y se describen los conceptos del mismo (ver **Figura 1**. Modelo ontológico).

Nomenclador: representa el contenedor de la información que se desea gestionar, está estructurado jerárquicamente por grupos e información.

Usuario: representa una persona o sistema informático que interactúa con el Nomenclador y el cual posee ciertos grados de accesibilidad, además es el encargado de introducir la información al sistema.

Administrador: representa el usuario encargado de la configuración, así como de la construcción de la estructura jerárquica (grupos).

Grupos: representa la raíz de la estructura jerárquica de un nomenclador, conformado a su vez por grupos e información (padre).

Información: representa el nivel final de la estructura jerárquica de un nomenclador (hijos).

Campo: representa los distintos atributos o características con los que puede contar un nomenclador.

Código: constituye un indicador que representa únicamente a un nomenclador específico. Este se construye a partir de la unión de los respectivos códigos de los grupos e información del nomenclador.

Asociar: operación que permite establecer si un campo será Agregado o Asignado a un nomenclador.

Agregado: clasificación que se le da a un campo cuando este solo va a pertenecer a un elemento padre.

Asignado: clasificación que se le da a un campo cuando este no va a pertenecer a un elemento padre y será heredado por sus hijos, formando así parte de estos últimos.

Estado: posibilita conocer el estado actual de la información en el sistema. Este puede ser, Activo o Pasivo.

Activo: representa el estado activo de la información. La información está siendo usada en el sistema.

Pasivo: representa el estado pasivo de la información. La información, aunque ha sido eliminada en el sistema, permanece físicamente en la base de datos.

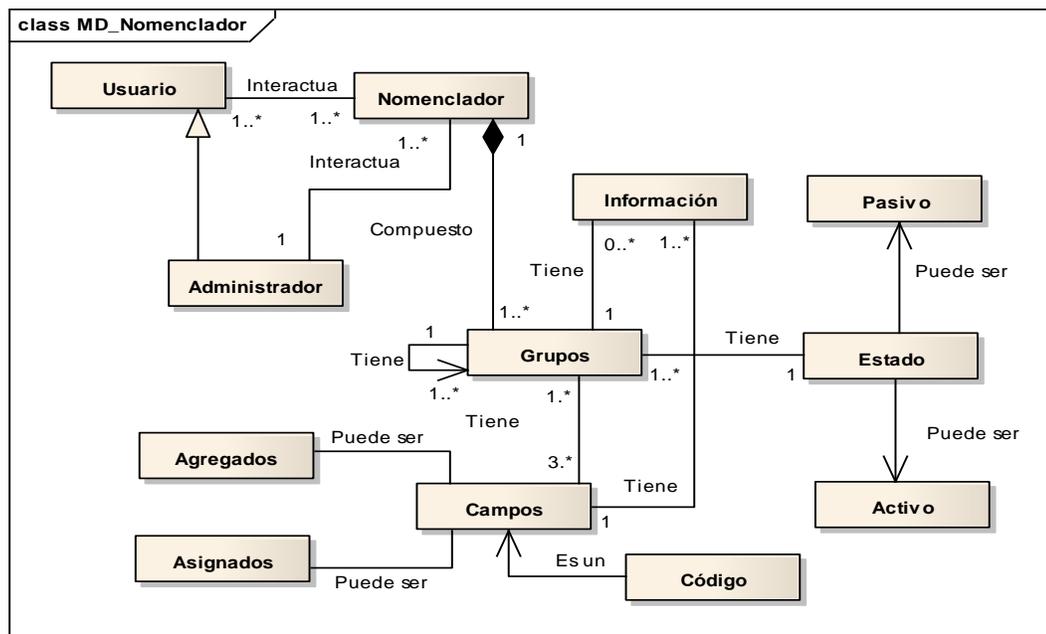


Figura 1. Modelo ontológico

Los términos usuario y administrador son puramente informáticos, se definen con el objetivo de identificar los posibles acceso a la aplicación. Los conceptos nomenclador, grupo, campo, código e información tienen una relación estrecha con lo que plantea Wurman en sus conceptos, específicamente con Ubicación, Alfabeto, Jerarquía y Categoría, permitiendo que un grupo pueda ser ordenado por cualquiera de los conceptos específicos, todo depende de la información que se esté almacenando en el sistema. Los conceptos campo, código e información, son características con que puede contar la información detallada.

En la investigación fue necesario introducir algunos conceptos importantes para manejar la información dentro del sistema. Hasta el momento los términos que maneja Wurman son importantes pero no suficientes, dejan fuera una parte importante de la administración del sistema como son, el estado que puede tener una información, como activo y pasivo, refiriéndose específicamente a si el usuario puede visualizarla o no en un momento determinado. De este modo se protege la información para los que no tienen autorización a cierto nivel del sistema. Una vez creados los campos en el sistema, se debe dar la posibilidad de que la información pueda ser cambiada y manejada según las necesidades del usuario y la flexibilidad que necesite el propio cliente; por lo que fue necesario introducir el término de asociar, combinado con agregar y asignar incorporándole a cada uno de los nomencladores los campos de información que van a llenar. Esto proporciona flexibilidad al sistema, haciéndolo adaptable para cualquier tipo de información, y toda aplicación informática que deba gestionar esta información los debe tener en cuenta a la hora de su concepción.

1.2 Situación problemática y objeto de automatización

Actualmente cada sistema posee la gestión de los requerimientos de los nomencladores que necesita, logrando que la información se mantenga aislada, descentralizada e independiente. Las soluciones actuales en algunos casos presentan implementaciones muy básicas, orientadas fundamentalmente a los nomencladores que gestionan. Así mismo, al estar descentralizada la gestión de estos requerimientos y como todas las aplicaciones no lo implementan de igual forma, no se puede crear una única estructura de manejar los nomencladores en todos los sistemas.

Existe información nomenclada a nivel internacional, lo que facilita el trabajo a las aplicaciones informáticas y a los usuarios de los sistemas, dictando la forma en que esta tiene que presentarse para su uso. Ejemplos de ella en salud son: el nomenclador Anatómico, Terapéutico, Químico (ATC)¹ y Formulario Nacional de Medicamentos (FNM)², entre otros (Barbado, 2011). Estos son usados en el Sistema para el control farmacológico (Synta), creado en la Facultad 7, así como en el Sistema de Gestión Hospitalaria (HIS, por sus siglas en inglés), también desarrollado en la Facultad 7, que hace uso también del nomenclador ATC. Ambos sistemas implementan, muestran y actualizan de forma distinta la misma información de este nomenclador. De esta manera los usuarios que trabajan con varios sistemas que cuenten con la misma información nomenclada, la visualizan de distintas formas. Esto trae consigo que le hagan rechazo a los sistemas informáticos y aquello que le podría ser de utilidad lo vean como un estorbo en su trabajo.

En los casos analizados, se encuentra que cada uno de los sistemas cuenta con su propia política o mecanismo de implementación de nomencladores, lo que puede comprometer la información. Si los sistemas en su creación no tienen en cuenta un adecuado desarrollo de los requerimientos que necesitan los nomencladores, para que al cambiar la información no se vea afectado todo el sistema, entonces corren el riesgo de tener tiempo de vida limitado. Si los sistemas no realizan la adecuada implementación de los requerimientos y la información que gestionan los nomencladores cambia en un tiempo determinado, los mismos se verían obligados a reimplementarse en su totalidad o gran parte de ellos.

A continuación se presenta un ejemplo del cambio que tendría que realizar un sistema y del tiempo que necesitaría en caso de usar sistemas que nomenclan información:

¹ATC: es un índice de sustancias farmacológicas y medicamentos, organizados según grupos terapéuticos.

²FNM: Información farmacológica de todos los medicamentos que se utilizan para la atención a la salud en Cuba

En la Facultad 7 en los años 2006-2008 se desarrolló un sistema llamado Balance Material, encargado de llevar la planificación de los materiales médicos gastables. El sistema necesitaba información nombrada como: materiales médicos gastables (teniendo en cuenta las características de cada uno de los grupos de materiales); las especialidades médicas, las unidades de salud, los profesionales y la ubicación de esas unidades. Los materiales médicos en una de las áreas (área de laboratorio, área de policlínica, etc.) pueden ser más de 5000. La implementación de esta información se hizo engorrosa en el sistema existiendo una tesis de pregrado solamente para nombrar los medicamentos y las especialidades médicas. Las mismas se dividían teniendo en cuenta la estructura que se presenta a continuación:

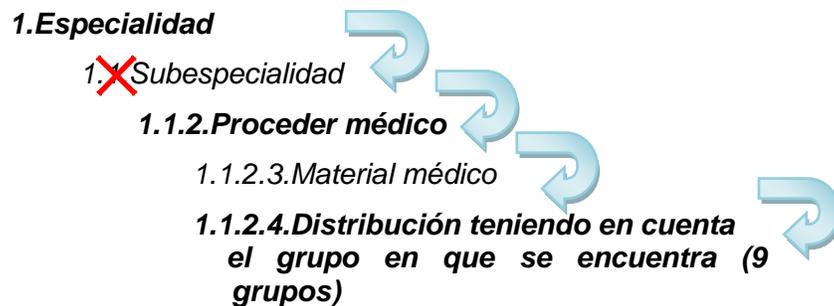


Figura 2. Estructura del nomenclador de especialidades médicas

Una vez terminado e integrado el nomenclador al sistema de Balance y Planificación, después de más de nueve meses de desarrollo, el Ministerio de Salud Pública (MINSAP) decidió que la estructura que tenían las especialidades médicas debía cambiar, eliminando el nivel de Subespecialidad, quedando de esta forma:

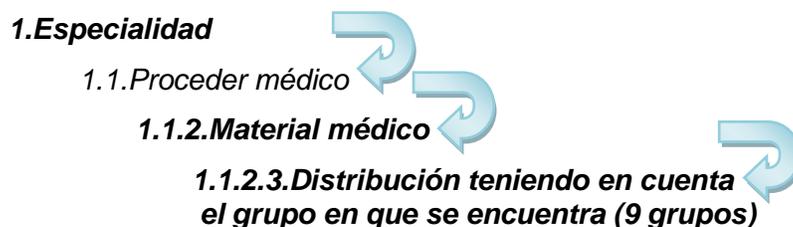


Figura 3. Estructura del nomenclador de especialidades médicas después de realizado el cambio

Este cambio en la estructura del nomenclador trajo consigo que un sistema que estaba a punto de desplegarse, se viera afectado en los nomencladores (la información interna del nomenclador cambió también la estructura), por lo que el sistema tuvo que implementarse casi en su totalidad. Toda la conexión interna del sistema estaba unida a la estructura que contenían los nomencladores (tablas de BD, clases de implementación, interfaz de usuario, etc.), por lo que todo tuvo que ser cambiado.

Este es un ejemplo de lo que puede causar en un sistema que no se tenga en cuenta en su desarrollo una buena gestión de la información nomenclada, porque todo lo que se podía haber adelantado en un futuro se vio afectado por el cambio de la información que no debería cambiar. Todo sistema que gestione información nomenclada y quiera disminuir el tiempo de desarrollo, debe tener una buena concepción de la información y buscar sistemas que sus requerimientos se centren solamente en gestionar esta información; por lo que entonces el sistema que se quiere desarrollar se debería preocupar nada más por la forma de consultar esa información que él necesita y que la almacena el otro sistema.

1.3 Ambiente de desarrollo propuesto

Se propone un conjunto de tecnologías y herramientas para el desarrollo del sistema, estas se describen a continuación.

1.3.1 Ontología

Para representar el conocimiento, las ontologías son la técnica más utilizada por sus características. Este término empezó a emplearse para referirse a las teorías de modelado del mundo y a los componentes de un sistema de conocimiento. Desde esta perspectiva, una ontología se define como “un conjunto de términos estructurados jerárquicamente para describir un dominio, que puede ser utilizado como el esqueleto fundamental para una base de conocimientos” (Arano, 2005) (Ferrí, 2009 pág. 31) (Gruber, 1993 pág. 5) (Noy, y otros, 2005).

En una ontología, “las definiciones asocian los nombres de las entidades en el universo de discurso (clases, relaciones, funciones u otros objetos) con texto legible desde un punto de vista humano, describiendo lo que significan los nombres, y axiomas formales que restringen las interpretación y el uso adecuado de estos términos” (Gruber, 1993 pág. 6). Una ontología facilita la explicación específica de los

conceptos (clases), sus propiedades y sus restricciones, así como la organización parcial de los conceptos a partir de sus relaciones. Por lo que no puede interpretarse que los conceptos se organizan exclusivamente de forma jerárquica. Una ontología es una tecnología conceptual que permite, definir el vocabulario para quienes necesitan compartir la información dentro de un determinado dominio. (Ferrí, 2009 pág. 31)

1.3.2 Modelo Vista Controlador (MVC)

Este patrón de arquitectura de software permite separar los datos de una aplicación, la interfaz de usuario y la lógica de negocio, en tres componentes distintos; esto proporciona múltiples vistas sobre un mismo modelo de datos. El patrón MVC se usa frecuentemente en aplicaciones web donde se utilizan diferentes interfaces de usuario y el código que provee los datos a la página, es dinámico. Dado que la vista se halla separada del modelo y no hay dependencia directa del modelo con respecto a la vista, la interfaz de usuario puede mostrar múltiples vistas de los mismos datos simultáneamente. Los tres elementos esenciales de este patrón son los siguientes: (Kicillof, y otros, 2004)

- *Modelo*: administra el comportamiento y los datos del dominio de la aplicación, responde a requerimientos de información sobre su estado usualmente formulados desde la vista, respondiendo a instrucciones, para cambiar el estado de los datos, habitualmente desde el controlador.
- *Vista*: presenta el modelo en un formato adecuado para interactuar, usualmente un elemento de interfaz de usuario.
- *Controlador*: responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista.

1.3.3 XML

Extensible Markup Language (XML) es un lenguaje muy simple que representa una manera distinta y más avanzada de hacer las cosas, cuya principal novedad consiste en permitir compartir los datos con los que se trabaja a todos los niveles, por todas las aplicaciones y soportes. El XML juega un papel importante en el mundo actual, que tiende a la globalización y la compatibilidad entre los sistemas. Es la tecnología que

permitirá compartir la información de una manera fácil, segura y fiable. Además, XML permite al programador dedicar sus esfuerzos a las tareas importantes cuando trabaja con los datos, permitiendo realizar algunos trabajos tediosos como la validación de estos o el recorrido de las estructuras, todo corre a cargo del lenguaje y está especificado por el estándar, el programador no tiene que preocuparse por ello. (Álvarez, 2001)

1.3.4 SOAP

En el núcleo de los servicios web se encuentra el Protocolo Simple de Acceso a Datos (SOAP por sus siglas en inglés), que proporciona un mecanismo estándar de empaquetar mensajes. SOAP ha recibido gran atención debido a que permite realizar Llamadas a Procedimiento Remoto (RPC por sus siglas en inglés) a un servidor web, bien en el cliente o en el servidor mediante el protocolo HTTP. Los mensajes tienen un formato determinado, empleando XML para encapsular los parámetros de la petición. Algunas de sus ventajas son: (González, 2004)

- No está asociado con ningún lenguaje.
- No se encuentra fuertemente asociado a ningún protocolo de transporte.
- Aprovecha los estándares existentes en la industria.
- Permite la interoperabilidad entre múltiples entornos.

1.3.5 WSDL

El Lenguaje de Descripción de Servicios Web (WSDL por sus siglas en inglés) permite que un servicio y un cliente establezcan un acuerdo en lo que se refiere a los detalles de transporte de mensajes y su contenido, a través de un documento procesable por dispositivos.

WSDL: (Booth, y otros, 2007)

1. Representa una especie de contrato entre el proveedor y el que solicita.
2. Especifica la sintaxis y los mecanismos de intercambio de mensajes.
3. Permite por separado la descripción de la funcionalidad abstracta de un servicio web a partir de los detalles concretos de cómo y dónde se ofrece la funcionalidad.

4. Una interfaz define la interfaz abstracta de un servicio web como un conjunto de operaciones abstractas, cada operación representa una simple interacción entre el cliente y el servicio.

1.3.6 Servicios Web XML

Los servicios web (Cabrera, y otros, 2004) tienen gran importancia en la tendencia de la computación distribuida, los estándares abiertos y el enfoque en la comunicación y la colaboración entre aplicaciones. Se ha creado un entorno donde los servicios Web XML están favoreciendo la plataforma para la integración de aplicaciones. Los servicios web funcionan como una aplicación “independiente” que ofrece una interfaz de acceso basada en mensajes, ajustándose a estándares para el desarrollo web. Estos cuentan con todos los beneficios propios de las tecnologías distribuidas en capas y constituyen un paso adelante en las aplicaciones basadas en componentes.

En la **Tabla 1**. Comportamiento de algunos indicadores de desempeño de los servicios web, se han representado algunas características de los servicios web atendiendo a varios indicadores de construcción y desempeño. (Ronda, 2004)

Tabla 1. Comportamiento de algunos indicadores de desempeño de los servicios web

Indicador	Desempeño de los servicios web
<i>Acoplamiento</i>	Débilmente acoplados, se basan en protocolos estándares y trabajan conjuntamente con independencia de dónde residen y/o cómo hayan sido implementados.
<i>Acceso</i>	Exponen su funcionalidad en estándares WSDL, accesibles a través de mensajes que se basan en protocolos abiertos e independientes de la plataforma.
<i>Control</i>	Son controlados y administrados de forma centralizada e independiente de las aplicaciones clientes.
<i>Ubicación</i>	Son transparentes a su localización en la red. A través del estándar WSDL se pueden conocer las particularidades de sus servicios y ubicación.

<i>Funcionamiento</i>	Habilitan direcciones URL que mediante el paso de una serie de parámetros, ejecutan algún proceso del negocio y devuelven información en distintos formatos (generalmente documentos XML).
<i>Rendimiento</i>	Su rendimiento sobre la red se ve afectado en ocasiones, debido a que todo el flujo de información es en formato de texto.
<i>Políticas de seguridad y tolerancia a fallos</i>	Son aplicaciones independientes que definen sus propias políticas de seguridad, configuración y escalabilidad. Ofrecen un conjunto de tecnologías adicionales para el manejo transaccional, la seguridad, entre otras.

1.3.7 Seguridad en aplicaciones web

La principal vulnerabilidad de la información en una aplicación web (PCMAG, 2010), se presenta en que la información viaja a través de la red, y puede ser sustraída por cualquier persona, no obstante, existen una serie de vulnerabilidades como la falsificación de identidad, el robo de sesiones autenticadas, la falsificación o modificación de los mensajes, entre otras. Una aplicación web que se diga segura, debe contemplar mecanismos que garanticen que efectivamente ninguna persona ajena al sistema, pueda modificar, obtener o falsificar datos de dicha aplicación.

Según esta guía, los principios básicos de seguridad que debe cumplir cualquier aplicación o servicio web son: (OWASP, 2005)

- **Validación de la entrada y salida de información:** la entrada y salida de información es el principal mecanismo de que dispone un atacante para enviar o recibir código malicioso contra el sistema. Por tanto, siempre debe verificarse que cualquier dato entrante o saliente es apropiado y está en el formato que se espera. Las características de estos datos deben estar predefinidas y debe verificarse en todas las ocasiones.
- **Utilización y reutilización de componentes de confianza:** debe potenciarse el principio de la reutilización en todos los casos en que sea posible. Por tanto, cuando exista un componente que resuelva un problema determinado de forma correcta, lo más adecuado es utilizarlo.

- **Proveer servicios de autenticación segura a las aplicaciones web:** las aplicaciones web deberán poseer reglas o mecanismos de autenticación, donde los certificados de seguridad establecidos puedan poseer variables que se modifiquen durante el tiempo en que la sesión se encuentre activa.
- **Verificación de privilegios:** los sistemas deben diseñarse para que funcionen con la menor cantidad posible de privilegios. Es importante que los procesos únicamente dispongan de los privilegios necesarios para desarrollar su función, de forma que queden compartimentados.
- **Ofrecer la mínima información:** ante una situación de error o una validación negativa, los mecanismos de seguridad deben diseñarse para que faciliten la mínima información posible. Estos mecanismos deben estar diseñados para que una vez denegada una operación, cualquier operación posterior sea igualmente denegada.

Se debe considerar además, que el desarrollo de aplicaciones web ha alcanzado un auge sin precedentes, pues las soluciones se orientan a fines y propósitos específicos. Esto a su vez, implica que las mismas manipulen información y datos de elevada sensibilidad, la cual podría encontrarse, sin la debida protección, en estados corruptos e inconsistentes, perdiendo así los principios básicos de la información, es decir, la confidencialidad, integridad y disponibilidad. (ISO27001, 2005)

1.4 Sistemas automatizados existentes asociados al campo de acción

Después de una búsqueda de sistemas informáticos que gestionan información poco variable en el tiempo, tanto en el ámbito internacional como nacional se evidenció que esta práctica se ha prolongado en el desarrollo de aplicaciones informáticas. Se puede evidenciar el uso de los mismos en varias ramas como la economía, la educación y tráfico, pero en la rama que más se ha usado es en la salud por la importancia de que todos los profesionales hablen el mismo idioma. Estos sistemas se han desarrollado sobre todo en el campo de la Web, con el objetivo de tener siempre en línea la información, logrando el fácil acceso y la actualización de la información. Aunque es válido señalar que algunos son de escritorio, bajo Licencia Pública General (GPL), permitiendo que los desarrolladores puedan mejorarlos para su buen uso en la sociedad.

Un estudio realizado demostró, que existe un número significativo de sistemas informático que gestionan nomencladores y estos son propietarios, aunque algunos de ellos hacen uso de servicios Web XML para que sistemas externos puedan consumir la información sin tener que comprar el sistema completo. A continuación se muestran algunos de los sistemas encontrados que en su desarrollo han tenido en cuenta la información poco variable en el tiempo. A nivel internacional se puede encontrar el Sistema para Prestadores en el área de economía, Nomenclador de Procedimientos y Servicios en el área de la medicina y Nomenclador cartográfico para personas con deficiencia visual que une dos ramas específicas, la medicina y el tráfico en la vía pública.

En el ámbito nacional se ha hecho uso de los mismos en diferentes ramas, pero en la medicina es donde más se utilizan, por las características particulares de la salud pública cubana. Algunos ejemplos son, el Registro de la Clasificación Internacional de Enfermedades (RCIE) y el Registro de Problemas de Salud de la Atención Primaria (RPSAP), entre otros.

1.4.1 Características de algunos sistemas internacionales

1.4.1.1 Sistema para Prestadores

DASUTeN (Dirección de Asociación Social de la Universidad Tecnológica Nacional) es un sistema informático que tiene como objetivo facilitar a los prestadores la generación y posterior envío del archivo que debe acompañar a cada facturación realizada a DASUTeN, y que se solicita como condición obligatoria para proceder al posterior pago de dicha facturación (Possetto, 2010). El sistema permite generar el archivo que DASUTeN solicita, evitando que se envíe la información de la forma tradicional. A través de internet se envía y se actualiza información, como el padrón de los afiliados activos y los nomencladores, a fin de evitar errores de datos en la generación del archivo, ayudando a los usuarios a tener el control de las facturas enviadas. (Pessano, 2010)

La opción “Nomencladores” muestra la clasificación de todos los productos (prestaciones o medicamentos) que podrán ser facturados a DASUTeN. De la misma manera que el padrón de afiliados, estos nomencladores se actualizarán con la frecuencia que sea necesaria, utilizando la opción “Actualizar” que se encuentra en el menú principal. (DASUTeN, 2010)

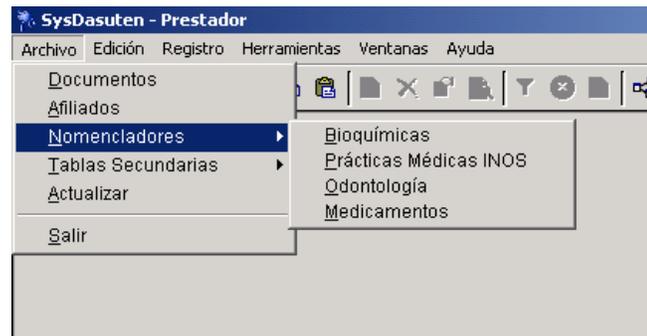


Figura 4. Creación de nomencladores en DASUTeN

Para prácticas no nomencladas (o alguna prestación omitida) habrá que solicitar la asignación de un código, el cual será cargado en DASUTeN y publicado para que la próxima vez que el prestador ejecute la opción “Actualizar”, tenga a su disposición el nomenclador actualizado. Esto es un proceso sencillo normalmente entendible casi en forma inmediata. (DASUTeN, 2010)

1.4.1.2 Nomenclador de procedimientos y servicios

El nomenclador de procedimientos y servicios es un documento técnico (.doc, .xls y .pdf) que consiste en el catálogo estandarizado de los procedimientos y servicios de salud disponibles en el mercado privado de salud, constituyendo un valioso instrumento dentro del Sistema Integrado de Información en Salud.

El nomenclador ha sido elaborado de manera conjunta por la Asociación de Clínicas Particulares (ACP), la Asociación Peruana de Empresas de Seguros (APESEG) y la Asociación Peruana de Entidades Prestadoras de Salud (APEPS). La Superintendencia de Entidades Prestadoras de Salud (SEPS) participó en su elaboración como facilitador y coordinador, en su calidad de principal usuario del nomenclador como estándar de procedimientos y servicios médicos.

El instrumento desarrollado se enmarca dentro de la propuesta de Transmisión Electrónica de Datos Estandarizados del Expediente de Facturación (TEDEF). Este tiene el propósito de contar con una base de datos estandarizada de información administrativa y económica, permanentemente actualizada, de manera que constituye un insumo importante para la toma de decisiones en todos los niveles de las organizaciones de salud.

El nomenclador detalla todos los procedimientos y servicios disponibles en el mercado privado de salud. No todos ellos son cubiertos por los financiadores. Los procedimientos y servicios que no tienen unidades están sujetos a condiciones pactadas entre financiadores y proveedores. (Particulares, 2008)

1.4.1.3 Nomenclador cartográfico para personas con deficiencia visual

Dos estudiantes de la Facultad de Matemática, Astronomía y Física de la Universidad Nacional de Córdoba, en Argentina, desarrollaron un software de distribución libre y gratuita, para facilitar el desplazamiento de los discapacitados visuales por la ciudad de Córdoba. El nomenclador brinda información a las personas con discapacidad visual para que puedan trasladarse de un punto a otro.

El programa se distribuirá bajo Licencia Pública General (GPL) para que pueda ser modificado, mejorado y ampliado por otros desarrolladores. Fue realizado en Java y tiene una arquitectura cliente-servidor. El servidor aloja la base cartográfica y procesa las solicitudes que recibe de los clientes. Otorgando mayor seguridad en la protección de los datos. Los autores consideran que en el futuro se deberán incorporar datos sobre el recorrido del transporte público y sumar tecnología de geoposicionamiento satelital. El nomenclador cartográfico, desarrollado durante los años 2007 y 2008, contiene las siguientes funcionalidades:

- Consulta sobre la existencia de una dirección particular en la ciudad de Córdoba.
- Consulta del camino más corto a pie entre dos puntos de la ciudad.

El resultado de una consulta se compone de:

- Mapa del camino a realizar que contiene colores en alto contraste para facilitar la creación de un mapa mental por parte del usuario con deficiencia visual parcial.
- Instrucciones en texto que guían al usuario sobre cómo llegar a su destino.
- Lectura audible de las instrucciones. Se brinda la posibilidad de guardar un archivo de sonido que podrá ser alojado en un celular o reproductor mp4, para su reproducción durante el trayecto. (Fernández, 2009)

1.4.2 Características de algunos sistemas nacionales

1.4.2.1 Registro de la Clasificación Internacional de Enfermedades (RCIE)

El módulo tiene como objetivo principal la adopción de un lenguaje unificado y estandarizado, propuesto por la Organización Mundial de la Salud (OMS) en la Décima Revisión de la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud (CIE-10). Todos los módulos que trabajen con diagnósticos tienen relación con este registro, que gestiona la codificación de todas las enfermedades, problemas relacionados con la salud que se puedan presentar en la atención médica diaria, causas de lesiones y muerte. Posteriormente se generan estadísticas de morbilidad, mortalidad y servicios, entre otros.

El *Editor* a nivel nacional de este módulo, es el único que puede agregar, modificar o eliminar los capítulos, grupos, categorías, subcategorías y problemas de salud que constan en la CIE, así como sus respectivas descripciones. Este registro tiene como valor agregado, que sirve como una especie de software educativo para el conocimiento de la CIE, en el cual el usuario a modo de *Visualizador* puede hacer búsquedas o leer todo lo relacionado con las diferentes enfermedades. (Díaz, 2009)



Figura 5. Registro de la Clasificación Internacional de Enfermedades (RCIE)

1.4.2.2 Registro de Problemas de Salud de la Atención Primaria (RPSAP)

El registro tiene como objetivo principal, gestionar una clasificación especial para los problemas de salud que se presentan en la Atención Primaria de Salud (APS), tratando que resulte una clasificación más breve y simple que la CIE-10 y además compatible con esta. Permite realizar búsquedas dinámicas de

acuerdo a criterios seleccionados por el usuario, ofreciendo un lenguaje unificado y estándar para la información de la morbilidad en la APS.

El *Editor* a nivel nacional, es el único que puede gestionar (Agregar, Modificar y/o Eliminar) la información (Nombre, Grupo y Descripción) de los Riesgos, Enfermedades y Discapacidades registradas, así como gestionar la información (Nombre y Tipo) de los Grupos y Sub-Grupos de discapacidades. El usuario *Visualizador* puede obtener el listado general de Grupos de Riesgos y Riesgos, Grupos de Enfermedades y Enfermedades, Grupos de Discapacidades y Discapacidades, Listado de Riesgos específicos por Grupos, Listado de Enfermedades específicas por Grupos, Listado de Discapacidades específicas por Grupos y detalles de códigos de CIE-10. (Díaz, 2009)



Figura 6. Registro de Problemas de Salud de la Atención Primaria (RPSAP)

1.5 Conclusiones parciales

Existe un conjunto de fundamentos teóricos conceptuales que constituyen la base para la correcta concepción del diseño e implementación de cualquier sistema informático. Estos no son suficientes, por lo que se introduce en la investigación una propuesta para todo sistema que empiece su desarrollo y los conceptos necesarios para lograr un buen desarrollo de los sistemas que gestionen información poco variable en el tiempo.

En el ámbito informático hay sistemas creados a la medida, con el objetivo de gestionar la información necesaria para saciar sus necesidades; dedicándose a recolectar, procesar y transmitir información específica; con herramientas, tecnologías y estructuras de desarrollo distintas; resolviendo un problema

puntual. Existe otro grupo de sistemas que en su desarrollo tuvieron en cuenta la integración, pero solo se centran en un conjunto de información; dejando fuera de este, un volumen de información que los sistemas externos necesitarían al integrarse

Los sistemas informáticos analizados no satisfacen las necesidades reales del desarrollo informático general, siendo todo lo planteado anteriormente un punto de partida en la investigación propuesta.

CAPÍTULO 2: PROPUESTA DE SOLUCIÓN

En este capítulo se realiza una descripción de la propuesta de solución, considerando para ello, la explicación de los usuarios implicados en el sistema, que contiene las principales definiciones que se utilizan en la propuesta y las relaciones que se establecen entre ellos. Se especifican los requerimientos funcionales que forman parte de la solución, se realiza una descripción de la arquitectura y se describe en términos de implementación, los servicios web a partir del problema que estos solucionan.

2.1 Descripción de la propuesta de solución

La propuesta de solución cuenta con dos actores fundamentales:

Administrador, tiene el permiso y la responsabilidad en el sistema de crear toda la estructura de los nomencladores a los cuales se les introducirá información (refiérase específicamente a la creación jerárquica de los grupos del nomenclador). Este debe contar con conocimientos informáticos, que le permitan realizar la configuración de todos los campos del nomenclador. Para ello debe saber qué tipo de información guardará (solo letras, letras y número, números decimales, etc.) y qué tipo de campo va a ser mostrado por el sistema (campo de texto, campo de texto multilínea, selección múltiple, lista menú, entre otros). Para poder terminar la configuración y creación de cada uno de los campos, el usuario debe conocer si el valor que va a guardar el campo puede ser nulo o no, con el objetivo de que el sistema pueda validar en caso de que el usuario deje ese campo vacío.

Usuario, actor fundamental de la propuesta de solución, una vez que el *Administrador* cree toda la estructura con que debe contar el nomenclador, el *Usuario* debe llenar dicha información haciendo uso de los campos asignados por el *Administrador*. Este usuario es el principal y más importante del sistema. El mismo tiene la responsabilidad de llenar el nomenclador con la información necesaria para que pueda ser usado por las aplicaciones que lo necesiten; en algunos negocios es más conocido como el experto en la información. En el momento de crear el *Usuario*, el *Administrador* del sistema le asigna los nomencladores a los cuales tendrá acceso.

Esta funcionalidad tiene dos objetivos específicos, el primero es que el usuario al entrar al sistema, pueda visualizar solamente los nomencladores a los que le dieron permiso en su creación, y el segundo es permitir que en el momento de consumir los nomencladores, haciendo uso de los servicios web, se pueda verificar que un usuario puede hacer uso de cierta información, de lo contrario se le niega el acceso a la misma.

Existirá un usuario externo (aplicación), el cual accederá al Sistema Nomenclador, contando con un usuario registrado en el mismo, identificándose en el momento de realizar la petición, con el objetivo que el Sistema Nomenclador sea capaz de identificar la lista de nomencladores a mostrarle, permitiendo que el usuario solamente pueda acceder a la información que le fue asignada. EL nomenclador devolverá la información al sistema que está realizando la petición a través de un XML, logrando la comunicación entre ambos sistemas de forma rápida

Ambas aplicaciones pueden coexistir en el mismo servidor si cuentan con la misma plataforma de trabajo, sino pueden desplegarse en servidores distintos. (Para mayor explicación de la forma de comunicación, ver el epígrafe **2.2 Implementación de la propuesta de solución**)

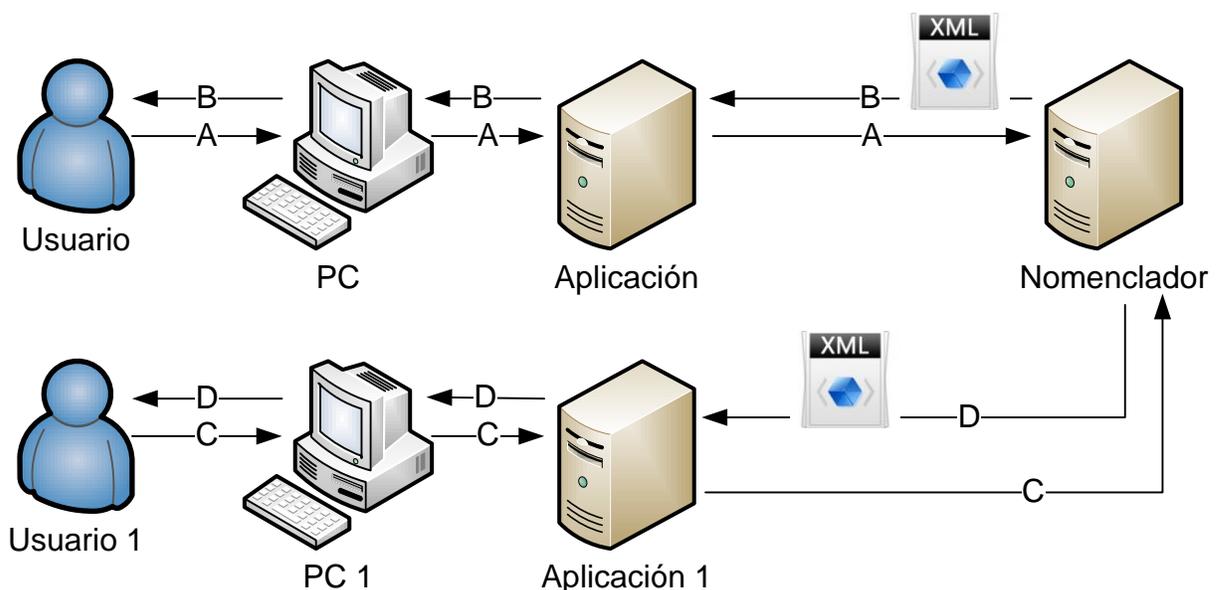


Figura 7. Acceso a la información del sistema

Para la creación un nomenclador en el sistema y la realización de cambios en el mismo, los usuarios deben seguir la estructura que se muestra a continuación, según sus responsabilidades:

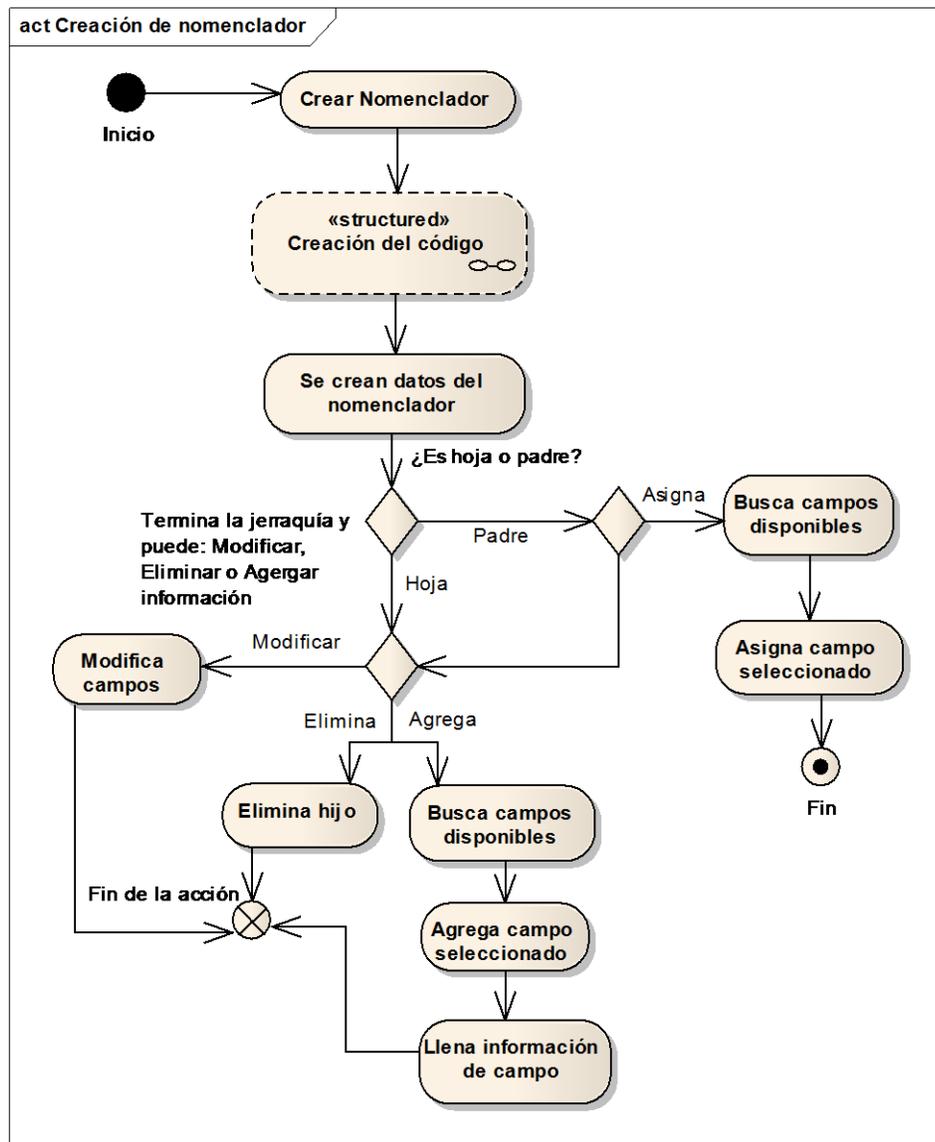


Figura 8. Diagrama de actividad del proceso *Crear Nomenclador*

La estructura de la **Figura 8.** Diagrama de actividad del proceso *Crear Nomenclador*, muestra detalladamente cómo se crea un nomenclador en el sistema, teniendo en cuenta las funcionalidades que

se pueden realizar si es un padre o un hijo lo que se ha creado. La actividad *Creación del Código* es estructurada, y sus características se explican más detalladamente en el epígrafe **2.2 Implementación de la propuesta de solución**, del presente capítulo. Ver **Anexo 1 y Anexo 3**.

2.1.1 Especificación de requerimientos de software

El sistema debe contar con características como la configuración y flexibilidad ante posibles cambios, por tanto este debe permitir:

- Gestionar los campos de los elementos nomencrados
- Gestionar el código de cada nomenclador
- Gestionar los grupos de los elementos a nomencrar
- Asociar los campos a los nomencladores
- Brindar servicios web que permitan el consumo de estos nomencladores por otras aplicaciones
- Gestionar la información relacionada con cada nomenclador
- Generar reportes de la información de un nomenclador específico o de varios de ellos
- Gestionar la información de los usuarios que interactúan con el sistema

Para poder cumplir con estas expectativas, se definió un conjunto de requerimientos, según la IEEE Standard Glossary of Software Engineering Terminology. Los requerimientos de software son condiciones o capacidades que deben estar presentes en un sistema o componente, para satisfacer un contrato, estándar, especificación u otro documento formal (Cháves, 2007 pág. 2). Es necesario hacer énfasis en la precisión con que se debe realizar esta tarea, por cumplir un papel primordial en el proceso de producción de software, pues se enfoca en un área fundamental: la definición de lo que se desea producir, permitiendo describir con mayor claridad el comportamiento del sistema, minimizando los problemas derivados de su desarrollo. A continuación se describen los principales requerimientos funcionales que posee la solución por cada uno de los usuarios definidos en el sistema.

Tabla 2. Requerimientos funcionales

Requerimientos funcionales	
Usuario de tipo <i>Administrador</i> :	
RF1 Insertar Grupo	RF10 Buscar Campo
RF2 Modificar Información Grupo	RF12 Insertar Estructura de Código
RF3 Eliminar Grupo	RF13 Modificar Estructura de Código
RF4 Buscar Grupo	RF14 Buscar Estructura de Código
RF5 Listar Grupo	RF15 Autenticación(<i>requisito que deben hacer ambos usuarios</i>)
RF6 Insertar Campo	RF16 Crear Usuario
RF7 Modificar Campo	RF17 Modificar Usuario
RF8 Eliminar Campo	RF18 Eliminar Usuario
RF9 Listar Campo	RF19 Buscar Usuario
RF11 Asociar Campo	RF20 Listar Usuario
Usuario de tipo <i>Usuario</i> :	
RF21 Crear Reporte	RF22 Insertar Información Grupo

2.1.2 Descripción de la arquitectura

Para el desarrollo del sistema se utilizó Symfony 1.4 como framework de desarrollo de aplicaciones web, el cual incluye el Mapeador de Objetos Relacional (ORM por sus siglas en inglés) Doctrine para el acceso a datos. Como lenguaje de programación Hypertext Pre-processor (PHP) 5.1. Como Sistema de Gestión de Base de Datos (SGBD) PostgreSQL 8.3 y para el manejo de datos, la herramienta pgAdmin. Las tecnologías seleccionadas para ser integradas en la solución, están basadas en software libre y de código abierto. También se empleó Enterprise Architect (EA) en su versión 7.1; como herramienta de modelado, apoyándose en el Lenguaje Unificado de Modelado (UML) 2.1, no siendo esta herramienta parte de la solución.

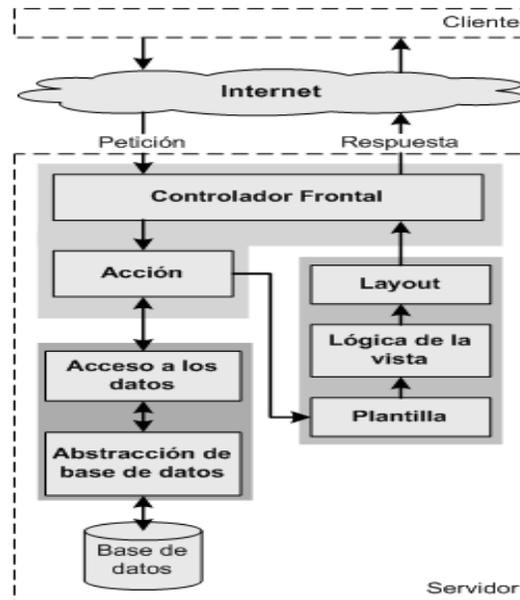


Figura 9. Estructura del framework Symfony

La utilización del framework Symfony permitió hacer uso del patrón MVC, pues está basado en este patrón de arquitectura. Este separa la lógica de negocio (el modelo) y la presentación (la vista), por lo que se consigue un mantenimiento más sencillo de las aplicaciones. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones (HTTP, consola de comandos, email, etc.). El modelo se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes por ejemplo el tipo de gestor de bases de datos utilizado por la aplicación.

En las aplicaciones web, el controlador generalmente se encuentra muy cargado, pues es el encargado de tareas como: el manejo de las peticiones del usuario y de la seguridad, cargar la configuración de la aplicación, entre otras actividades.

Por tal motivo, el controlador en Symfony normalmente se divide en un Controlador Frontal, que es único para cada aplicación, y las Acciones, que incluyen el código específico del controlador de cada página. Una de las principales ventajas de utilizarlo, es que ofrece un punto de entrada único para toda la aplicación. Así, en caso de que sea necesario impedir el acceso a la aplicación, solamente es necesario editar el script correspondiente (Potencier, y otros, 2008), ventaja que se utiliza en el sistema, para consumir la información a través de los servicios web.

En la Vista también se puede aprovechar la separación del código, por lo que Symfony la separa en *layouts* (capas externas), *templates* (plantillas) y *lógica de la vista*:

- *layouts*: es la parte que permanece invariable para las páginas de la aplicación.
- *templates*: solo se encargan de visualizar las variables definidas en el controlador.
- *lógica de la vista*: se utiliza la librería Ext JS, que incluye un conjunto de componentes que aportan mejoras visuales y facilitan un mejor intercambio con el sistema.

La capa del *Modelo* se puede dividir en la capa de Acceso a los datos y en la capa de Abstracción de la base de datos. De esta forma, las funciones que acceden a los datos, no utilizan sentencias ni consultas que dependen de una base de datos, sino que utilizan funciones escritas en el lenguaje propio del ORM Doctrine. Si se cambia de sistema gestor de base de datos (PostgreSQL en el caso particular del sistema), solamente es necesario actualizar la capa de Abstracción de la base de datos. El acceso a datos se realiza mediante el uso del motor de persistencia Doctrine; que abstrae al sistema del uso de cualquier sistema gestor de base de datos, mediante el uso de la programación orientada a objetos. Haciendo posible tratar las tablas como objetos del sistema. (Potencier, y otros, 2008)

2.2 Implementación de la propuesta de solución

La implementación es el centro de las iteraciones durante la fase de construcción, su organización se realiza de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación, y el lenguaje de programación utilizado; así como la dependencia que se establece entre estos componentes. (Pressman, 2005)

En términos de implementación, la creación del código se considera uno de los procesos más importantes del funcionamiento del Sistema Nomenclador, permitiéndole que sea configurable y adaptable a cualquier entorno de trabajo. En la siguiente figura se muestra el diagrama de actividad del proceso Crear código dentro del sistema.

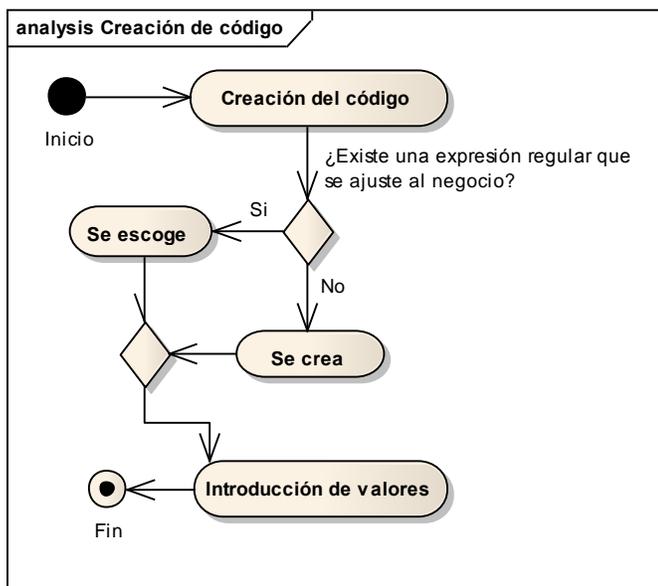


Figura 10. Diagrama de actividad del proceso Crear código (ver **Anexo 2**)

La estructura del código tiene un valor significativo en dependencia del tipo de información para el cual se use. Existen códigos que son los que definen la información, permitiendo al usuario realizar una búsqueda más rápida y así definir en qué nivel o subnivel se encuentra la misma; otros simplemente se usan con el objetivo de enumerar la información que se maneja. En muchos nomencladores, el código es una información muy importante y permite que toda forma de comunicación sea a través de él. Una estructura de código que contiene información adjunta está presente en el nomenclador Anatómico Terapéutico y Químico (ATC), sistema que se utiliza para la clasificación de las drogas a nivel internacional y es muy usado por los médicos cubanos.

El sistema de clasificación divide las drogas en diferentes grupos, según el órgano o sistema en el que actúan y/o sus características terapéuticas y químicas. Cada código a nivel de ATC es sinónimo de una sustancia farmacéutica utilizada en una sola indicación (o uso). Esto significa que un medicamento puede tener más de un código: el ácido acetilsalicílico (Aspirina), por ejemplo, es A01AD05 como una droga para el tratamiento oral local, B01AC06 como un inhibidor de plaquetas y N02BA01 como analgésico y antipirético. Por otro lado, diferentes marcas comparten el mismo código si tienen el mismo principio activo e indicaciones. En este sistema, las drogas se clasifican en grupos de cinco niveles diferentes:

El primer nivel del código indica el grupo anatómico principal y contiene 14 grupos principales:

Tabla 3. Grupos anatómicos principales

A	Aparato digestivo y Metabolismo
B	Sangre y órganos hematopoyéticos
C	Aparato cardiovascular
D	Dermatología
G	Aparato genitourinario y hormonas sexuales
H	Hormonas de uso sistémico, excepto las sexuales
J	Antifécciosos por vía general
L	Terapia antineoplásica
M	Aparato músculo esquelético
N	Sistema nervioso central
P	Parasitología
R	Aparato respiratorio
S	Órganos de los sentidos
V	Varios

El segundo nivel del código indica el grupo terapéutico principal y consta de dos dígitos. El tercer nivel del código indica el subgrupo terapéutico/farmacológico. El cuarto nivel indica el subgrupo químico /terapéutico/farmacológico y el quinto nivel el nombre del principio activo. (Saladrigas, 2004)

Tabla 4. Estructura del código ATC de la Metformina

Niveles		Código ATC	Descripción
1	Grupo anatómico principal	A	Tracto alimentario y metabolismo
2	Subgrupo terapéutico	A <u>10</u> (segundo nivel)	Fármacos usados en diabetes
3	Subgrupo farmacológico	A10 <u>B</u> (tercer nivel)	Fármacos hipoglucemiantes orales
4	Subgrupo químico	A10B <u>A</u> (cuarto nivel)	Biguanidas

5	Nombre de la sustancia farmacéutica (principio activo).	A10BA <u>02</u> (quinto nivel)	Metformina
---	---	--------------------------------	------------

El código de estos medicamentos es conocido por los profesionales que trabajan con dicha información a nivel internacional y son capaces de saber a través del código, qué medicamento es.

La creación de un código flexible como se muestra en el ejemplo anterior, es una de las funcionalidades que se define como crítica en el sistema, permitiendo que cada uno de los niveles en que se crea la información cuente solamente con su estructura; formando en la unión de cada nivel, el código final con que contará la información. La estructura del código se logra usando *Expresiones regulares* (Marco, 2011) definidas para cada uno de los niveles desde el padre (raíz del nomenclador) hasta la última hoja, que es la que gestiona la unión de todos los hermanos y la que cuenta con la estructura del código final.

El código de la *Metformina*, haciendo uso de las expresiones regulares, quedaría de esta forma en el sistema:

Tabla 5. Estructura regular del código ATC de la Metformina

Código ATC	Expresiones regulares
A	[A-Z]
A <u>10</u> (segundo nivel)	[0-9] [0-9]
A10 <u>B</u> (tercer nivel)	[A-Z]
A10BA <u>A</u> (cuarto nivel)	[A-Z]
A10BA <u>02</u> (quinto nivel)	[0-9] [0-9]
Expresión regular del medicamento <i>Metformina</i>	[A-Z] [0-9] [0-9] [A-Z] [A-Z] [0-9] [0-9]

Para crear la estructura del código, diferente por cada nomenclador, fue necesario desarrollar un método que implementara expresiones regulares diferentes en dependencia de las características que presenta cada información. Para la creación de la expresión regular de cada fragmento del código, se tienen en cuenta todos los valores que cada fragmento puede tomar:

[A-Z] Considera coincidencia: cualquier dígito comprendido entre A y Z (ambos inclusive), mayúscula.

[a-z] Considera coincidencia: cualquier dígito comprendido entre a y z (ambos inclusive), minúscula.

[0-9] Considera coincidencia: cualquier dígito comprendido entre 0 y 9 (ambos inclusive).

[a-zA-Z0-9] Considera coincidencia: cualquier dígito comprendido entre a y z (ambos inclusive), minúscula, A y Z (ambos inclusive) mayúscula y 0 y 9 (ambos inclusive). De esta forma se garantiza que los caracteres del código que se desee formar, puedan tomar cualquier valor.

Para validar la estructura del código se usa la función *Verifica*, la cual realiza una búsqueda en la cadena por medio de la expresión regular incluida en el patrón; si son iguales devuelve el valor booleano TRUE, en caso contrario devuelve FALSE. Este objetivo se logra haciendo uso de la función predefinida *preg_match*, para conocer la existencia o inexistencia de texto dentro de una cadena o de un fichero, a través de los parámetros (patrón, cadena).

La validación del código permite verificar que el valor introducido en el campo *Código*, en el momento que se crea el nomenclador, coincida con los valores que tiene la expresión regular creada. Esto ayuda al usuario a no confundirse en el momento de llenar el campo. En caso de no acordarse de los valores que debe introducir, el sistema mostrará un mensaje indicando qué debe entrar el usuario.

2.3 Definición de los servicios web implementados

Para la comunicación con el sistema fue necesario crear varios servicios web y a su vez los WSDL (W3C, 2010) (Ronda, 2004). Estos elementos contienen algunas pautas, de modo que sea posible cierta estandarización y los usuarios finales del Sistema Nomenclador puedan familiarizarse con ellos. Brindan la posibilidad, a componentes o sistemas externos, de utilizar la información que gestiona independientemente de la plataforma o lenguaje en el que hayan sido desarrollados.

A continuación se describen estos servicios, haciéndose énfasis en los parámetros de entrada y las estructuras de salida que utilizan. La clase *sp_actions.class.php*, es la encargada de gestionar toda la información acerca de los servicios web que brinda el nomenclador a los distintos sistemas clientes. Cuenta con funcionalidades que devuelven un nomenclador, (dado su identificador y su nombre) y todos los hijos (dado el identificador de un nomenclador padre).

2.3.1 Información de un nomenclador dado su ID en el sistema

Este servicio web brinda a los usuarios externos toda la información de un nomenclador dado su ID (identificador) correspondiente en el sistema. A continuación se muestra la descripción del servicio web, así como sus parámetros de entrada y salida.

Tabla 6. Descripción de los parámetros de entrada y salida del método *GetNomencladorDadold*

Nombre del servicio: GetNomencladorDadold	
Nombre de la función: getNomencladorDadold()	
Parámetros de entrada	
Descripción	Tipo de dato
sld	int
Parámetros de salida	
Descripción	Tipo de dato
Id	int
Nombre	string
Código	string
Descripción	string
ArrayCampos	array

Dado un ID como parámetro, la función devuelve el nomenclador con su información asociada; en caso de no encontrar nomenclador que coincida con la búsqueda devuelve un objeto vacío. Los parámetros que devuelve del nomenclador encontrado son: *Nombre* del nomenclador, *Código* con que cuenta el nomenclador, *Descripción* indicando para qué puede ser usada la información que contiene el nomenclador y los *Campos* con su valor asociado.

2.3.2 Información del nomenclador dado un nombre

Este servicio web le posibilita al usuario conocer, dado el nombre de un nomenclador, toda la información asociada a este; en caso de existir más de uno con el mismo nombre, el servicio le mostrará la información de todas las coincidencias. A continuación se muestra la descripción del servicio web, así como sus parámetros de entrada y salida.

Tabla 7. Descripción de los parámetros de entrada y salida del método *GetNomencladorLikeNombre*

Nombre del servicio: GetNomencladorLikeNombre
Nombre de la función: getNomencladorLikeNombre()

Parámetros de entrada	
Descripción	Tipo de dato
sNombre	string
Parámetros de salida	
Descripción	Tipo de dato
Id	int
Nombre	string
Código	string
Descripción	string
ArrayCampos	array

La función devuelve los nomencladores con su información asociada, dado un nombre pasado como parámetro, en caso de no encontrar ningún nomenclador que coincida con la búsqueda, devuelve un objeto vacío.

2.3.3 Lista de hijos dado el identificador del padre

Este servicio permite a los usuarios conocer la lista de hijos y sus valores, pasando como parámetro el identificador del padre. De esta forma se puede conocer la información asociada y asignada a los hijos.

Tabla 8. Descripción de los parámetros de entrada y salida del método *GetHijosDadoldPadre*

Nombre del servicio: GetHijosDadoldPadre	
Nombre de la función: GetHijosDadoldPadre	
Parámetros de entrada	
Descripción	Tipo de dato
sID	int
Parámetros de salida	
Descripción	Tipo de dato
Id	int
Nombre	string
Código	string
Descripción	string
ArrayCampos	array

Tabla 9. Descripción de los parámetros de entrada y salida del método *CampoValor*

Nombre del servicio: GetHijosDadoldPadre	
Nombre de la función: CampoValor	
Parámetros de entrada	
Descripción	Tipo de dato

Parámetros de salida	
Descripción	Tipo de dato
nombrcampo	string
valorcampo	string

Esta función devuelve el nombre de los campos que tienen los hijos del nomenclador consultado, así como sus respectivos valores; permitiendo al usuario acceder directamente a la información del nomenclador: último nivel de jerarquía que este presenta.

2.3.4 Visión integrada de los resultados de implementación

A continuación se describe, mediante un diagrama de actividades, la visión integrada de los resultados de implementación (Ver **Figura 11**. Visión integrada de los resultados de implementación). En este diagrama se muestra cómo el sistema maneja la integración con otras aplicaciones y se describe cómo se adquiere la información, utilizando de forma integrada los servicios implementados. La secuencia general del proceso de consumo de información es la que sigue:

Paso 1. Identificar qué tipo de información necesita el usuario consumir, y determinar posteriormente el servicio que esté más acorde a sus objetivos.

Paso 2. Realizar la petición mediante el servicio *GetHijosDadoldPadre*, *GetNomencladorLikeNombre*, *GetNomencladorDadold*, basándose en la documentación que quiere obtener.

Paso 3. Verificar que el usuario tiene permiso para acceder a la información solicitada, usando la función *executeVerificarUsuarioNomenclador*.

Paso 4. Integrar al sistema la información obtenida.

El **Paso 3** es el más importante, teniendo en cuenta la responsabilidad que tiene de permitir el acceso a la información solicitada por un cliente. Todo el proceso de verificación se realiza entre la *Aplicación* y el Sistema Nomenclador. En el momento del pedido, se valida que el usuario tenga permiso, haciendo uso de la función *executeVerificarUsuarioNomenclador*, se busca el usuario en el sistema así como la lista de nomencladores a la que tiene permiso y se verifica la misma con el pedido que está realizando; en caso

de tener acceso se responde positivamente, devolviendo el XML; en caso de no tener permiso se envía un mensaje de error. Esto proporciona que cada usuario solamente pueda hacer uso de la información que le corresponde.

Es necesario tener en cuentas algunos consejos para consumir los servicios web que brinda la solución:

1. Analizar las necesidades de integración del componente que se esté desarrollando con los componentes ya desarrollados o en fase de desarrollo.
2. Identificar la información que será utilizada, es decir, aquellos datos que el componente en desarrollo almacenará en su base de datos, pero que la obtiene consumiendo servicios del Nomenclador.
3. Realizar un levantamiento de los métodos mediante los cuales será modificada o eliminada esta información reutilizada, teniendo en cuenta además, el tipo de dependencia o restricción que se establece.
4. Implementar los métodos en dependencia de las necesidades de los componentes, realizando las consultas necesarias.

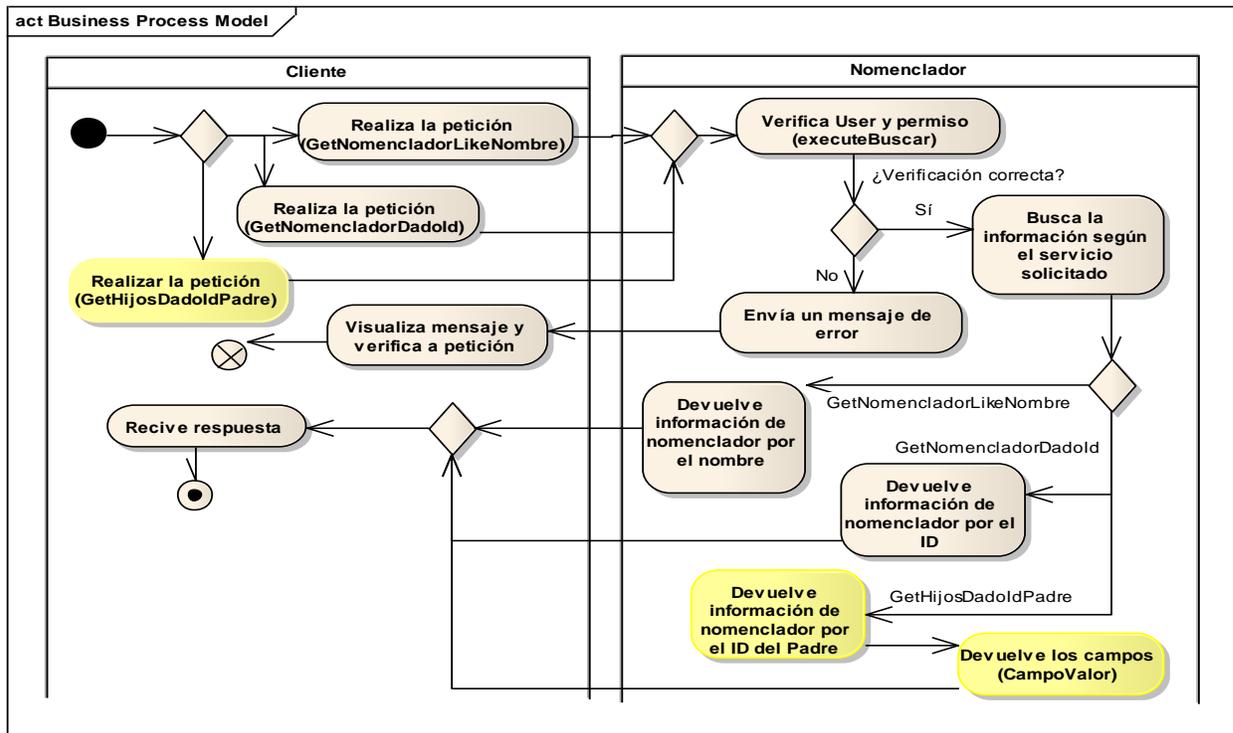


Figura 11. Visión integrada de los resultados de implementación

Se puede concluir que el Sistema Nomenclador es completamente extensible a las aplicaciones que lo necesiten (ver **Figura 11**. Visión integrada de los resultados de implementación). El autor pretende mostrar de forma explícita cómo se integran todos los servicios implementados para proveer, a los usuarios finales, la información solicitada.

2.4 Conclusiones parciales

La propuesta de solución, representa una aplicación configurable y extensible. En este capítulo se describió la arquitectura sobre la cual se desarrolló el sistema, las principales funcionalidades que presenta, los principales servicios web implementados y su forma de consumo.

CAPÍTULO 3: ANÁLISIS Y VALORACIÓN DE LOS RESULTADOS

El Sistema Nomenclador forma parte de los resultados de la investigación, este permite la gestión de todos los requerimientos de la información poco variable en el tiempo, para aquellos sistemas externos que consuman sus servicios, lográndose de esta forma la reutilización del código y evitándose que se realicen acciones innecesarias fuera de cada negocio. Esto se logra mediante la publicación de servicios Web XML debidamente descritos, utilizando los WSDL.

3.1 Validación de los resultados de integración

3.1.1 Pruebas de integración

Para la validación de la información se integró el Sistema Nomenclador a cuatro sistemas diferentes, los cuales se listan a continuación:

- *Sistema de Balance y Planificación de Insumos Médicos (alas BAP)*
- *Sistema para el control de Reacciones Adversas al Medicamento (RAM)*
- *Sistema para el Consumo de Medicamentos (CM)*
- *Sistema para el Control de Recetas Médicas (CRM)*

Estos sistemas cuentan con un conjunto de requisitos funcionales y una parte de ellos es utilizada para demostrar el resultado de la investigación.

Para el desarrollo de las pruebas se utilizó una computadora personal Haier, con un GB de Memoria de Acceso Aleatorio (RAM por sus siglas en inglés) y un procesador Core Duo a 1.87 Ghz de frecuencia, donde fue desplegado el Sistema Nomenclador completamente, incluyendo su base de datos. Se debe tener en cuenta, que los WSDL de estos servicios fueron desarrollados mediante la biblioteca que proporciona Symfony, *ckWebServicePlugin*. Esta brinda la posibilidad de analizar si los WSDL están correctamente implementados, así como construir peticiones SOAP al servidor, predefiniendo inclusive los tiempos máximos de respuesta.

CÁPITULO 3: ANÁLISIS Y VALORACIÓN DE LOS RESULTADOS

En la **Tabla 10**. Requisitos funcionales por sistemas, se muestra la lista de los sistemas y la cantidad de Requisitos Funcionales (*RF*) y Requisitos Funcionales Nomenclados (*RFN*) con que cuentan, así como el porcentaje de RFN sobre el total de RF.

Tabla 10. Requisitos funcionales por sistemas

Sistemas	Total de RF	RF	RFN	%
alas BAP	59	39	20	33.89
RAM	92	27	65	70.65
CM	57	21	36	63.15
CRM	35	26	9	25.71

La mitad de los requisitos funcionales nomenclados de cada sistema sobrepasan el 50 por ciento del total de requisitos de software, mostrándose en la gráfica de la **Figura 12**. Representación gráfica de los requisitos funcionales por sistema.

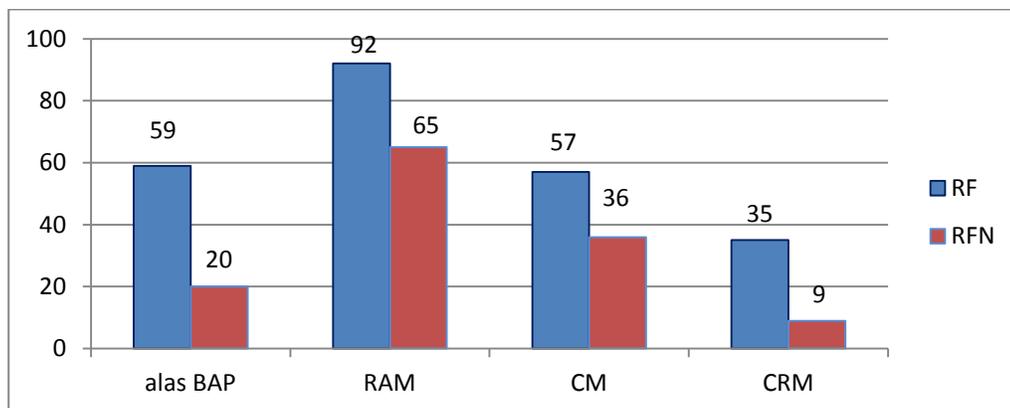


Figura 12. Representación gráfica de los requisitos funcionales por sistema

En la **Tabla 11**. Tiempo total de implementación por sistema, se evidencia el tiempo de implementación de los RFN expresado en meses:

Tabla 11. Tiempo total de implementación por sistema

Sistemas	Total de RF	RF	Tiempo I	RFN	Tiempo I	Tiempo Total
alas BAP	59	39	<u>3</u>	20	<u>5</u>	8
RAM	92	27	<u>6</u>	65	<u>10</u>	16
CM	57	21	<u>6</u>	36	<u>4</u>	10

CÁPITULO 3: ANÁLISIS Y VALORACIÓN DE LOS RESULTADOS

CRM	35	26	<u>4</u>	9	<u>1</u>	5
-----	----	----	----------	---	----------	----------

Los equipos de desarrollo que crearon estos sistemas conocían las herramientas de trabajo y el lenguaje de implementación. En las tablas que se muestran a continuación (**Tabla 12.** Entorno de desarrollo de los sistemas y **Tabla 13.** Tiempo de integración de los sistemas expresado en meses) se expone el lenguaje de programación, gestor de base de datos y framework que se utilizaron en el desarrollo de cada sistema y el tiempo que tardaron los equipos de desarrollo en integrar los mismos al Sistema Nomenclador.

Tabla 12. Entorno de desarrollo de los sistemas

Sistemas	Lenguaje	BD	Framework
alas BAP	PHP 5.1	MySql Server 5.0	-
RAM	PHP 5.1	Postgre 8.4	Symfony 1.4
CM	PHP 4.0	Postgre 8.4	CodeIgniter
CRM	PHP 5.1	Postgre 8.4	Symfony 1.1

Tabla 13. Tiempo de integración de los sistemas expresado en meses

Sistemas	Integración (días)
alas BAP	<u>30</u>
RAM	<u>60</u>
CM	<u>15</u>
CRM	<u>10</u>

Al determinarse las necesidades de los sistemas y luego de realizada su integración con el Sistema Nomenclador (utilizando los tres servicios), se verificó que se disminuyó el tiempo de desarrollo de los mismos.

Tabla 14. Tiempo total de la implementación teniendo en cuenta la integración

Sistemas	Antes de integrar(meses)	Tiempo de implementación + integración (meses)	Tiempo de implementación + integración (días)	% de <i>disminución de desarrollo</i>
alas BAP	<u>8</u>	<u>3.5</u>	<u>84</u>	56,3
RAM	<u>16</u>	<u>8</u>	<u>192</u>	50

CM	<u>10</u>	<u>7</u>	<u>168</u>	30
CRM	<u>5</u>	<u>2</u>	<u>48</u>	60

El tiempo promedio de desarrollo disminuyó en un 49 por ciento. Se logró con la integración una disminución mayor que el 50 por ciento en la mayoría los sistemas (los datos expuestos anteriormente se muestran en la **Figura 13**. Representación gráfica de los tiempos de desarrollo teniendo en cuenta la integración).

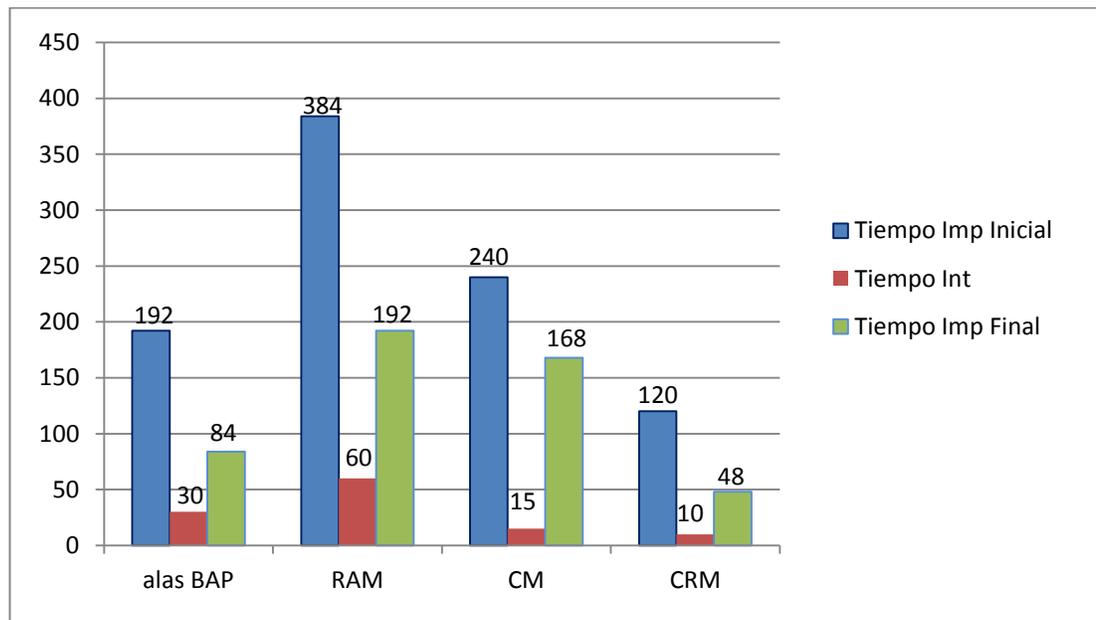


Figura 13. Representación gráfica de los tiempos de desarrollo teniendo en cuenta la integración

3.2 Encuesta aplicada a desarrolladores y especialistas participantes en la integración de la solución

Se aplicó una encuesta de seis preguntas, contando esta con 39 puntos internos, de ellos pueden ser marcados como factibles 15, no factibles 19 y cinco indecisos (“No sé”) (ver **Anexo 4**). Dicha encuesta se aplicó a una muestra de 15 ingenieros (ver **Anexo 5**), dentro de ellos: dos jefes de los departamentos productivos del Centro de Informática Médica de la Facultad 7, un jefe de proyecto y 12 desarrolladores de

los sistemas que se integraron al Sistema Nomenclador. Los resultados arrojados se muestran en la **Figura 14**. Grado de aceptación del sistema nomenclador y se describen después de esta.

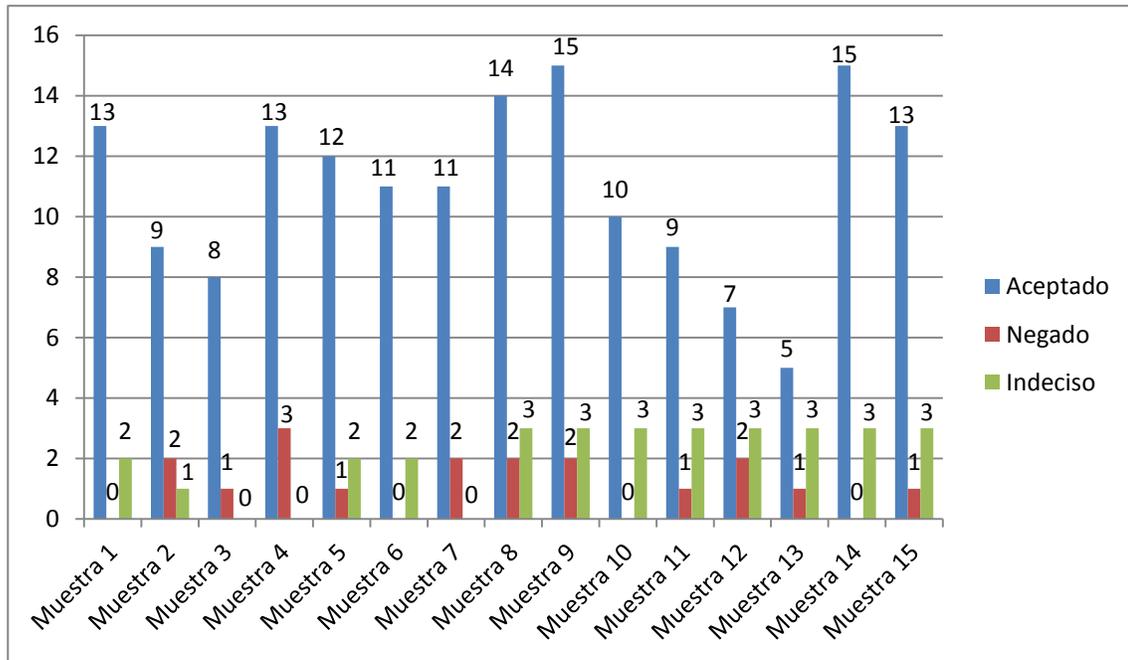


Figura 14. Grado de aceptación del Sistema Nomenclador

En la gráfica anterior se evidencian los niveles de aceptación, de negación y de indecisión, que tuvieron las personas encuestadas respecto al Sistema Nomenclador. El grado de aceptación fue mayor que el de negación y hubo un pequeño porcentaje de indecisión. De 15 puntos factibles, fueron seleccionados en algunos casos el máximo y el 66,6 por ciento seleccionó más de 10 puntos de aceptación.

La encuesta arrojó también, que el 80 por ciento de los encuestados están de acuerdo con que el uso del sistema disminuye el tiempo de desarrollo de las aplicaciones que lo utilicen; mostrándose dicha información en la gráfica de la **Figura 15**. Aumento del desarrollo de las aplicaciones informáticas que gestionan nomencladores. La mayoría de los encuestados creen que la información debe estar centralizada en una sola aplicación, o por lo menos la información que necesiten los sistemas que manejen información semejante, mostrándose dicha información en la **Figura 16**. Centralización de la información en un solo sistema informático.



Figura 15. Aumento del desarrollo de las aplicaciones informáticas que gestionan nomencladores



Figura 16. Centralización de la información en un solo sistema informático

A través de la encuesta se pudo comprobar que la mayoría de los desarrolladores creen que invierten demasiado tiempo implementando requisitos que no son propiamente de su negocio, trayendo como consecuencia que el personal dedicado a esta tarea se desgaste en actividades que no le corresponden. En la siguiente figura se evidencia cómo la mayoría de los encuestados creen que realmente este desarrollo (de los requisitos funcionales nomenclados) les proporciona una pérdida de tiempo.

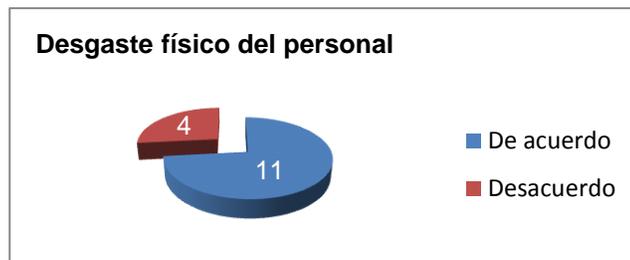


Figura 17. Desgaste físico del personal que desarrolla sistema informáticos que gestionan nomencladores

De los 15 encuestados: 12 participaron como desarrolladores en la integración con el Sistema Nomenclador y 13 identifican la factibilidad que proporciona realizar una integración y no desarrollar todos los nomencladores que puede presentar un sistema, permitiéndole la integración y la disminución del tiempo de desarrollo de las aplicaciones.



Figura 18. Análisis de la factibilidad de la integración por parte de los desarrolladores

La encuesta aplicada provee una fuente directa de información, que aporta un grado de madurez al desarrollo de la solución propuesta.

3.3 Aplicación de la técnica IADOV para medir satisfacción

Se aplicó la *técnica* de **IADOV** (Vélez, 2007 pág. 30) (Rodríguez, y otros, 2002) para obtener el grado de satisfacción de los desarrolladores de los sistemas que se integraron con el Sistema Nomenclador. Esta es una vía indirecta para el estudio de la satisfacción. Los criterios utilizados se fundamentan en las relaciones que se establecen entre tres preguntas cerradas, que se intercalan dentro de la encuesta aplicada en el epígrafe anterior.

Para obtener los resultados de la aplicación de la técnica, es necesario conocer la escala de satisfacción, así como la fórmula para determinar la satisfacción del grupo.

La escala de satisfacción personal e individual para estas actividades responde la siguiente estructura en función de la puntuación obtenida:

La escala de satisfacción

1. Clara satisfacción

CÁPITULO 3: ANÁLISIS Y VALORACIÓN DE LOS RESULTADOS

2. Más satisfecho que insatisfecho
3. No definida
4. Más insatisfecho que satisfecho
5. Clara insatisfacción
6. Contradictoria

Esta técnica permite establecer una referencia grupal permitiendo calcular el índice de satisfacción grupal de los desarrolladores participantes en la integración con el Sistema Nomenclador. Para poder ponderar este ISG (Índice de Satisfacción Grupal), se establece una escala numérica entre +1 y -1 de la siguiente manera.

+1	Máximo de satisfacción
0,5	Más satisfecho que insatisfecho
0	No definido y contradictorio
-0,5	Más insatisfecho que satisfecho
-1	Máxima insatisfacción

La satisfacción grupal se calcula por la siguiente fórmula:

$$ISG = \frac{A(+1) + B(+0,5) + C(0) + D(-0,5) + E(-1)}{N}$$

$$ISG = \frac{11(+1) + 2(+0,5) + 1(0) + 1(-0,5) + 0(-1)}{15} = 0,76$$

En la **Tabla 15** se muestra el *Cuadro lógico de V.A. IADOV*, donde se realiza la triangulación de las preguntas directas.

Tabla 15. Cuadro lógico de V.A. IADOV

2. ¿Al desarrollar un sistema informático tiene en cuenta la estructura de la información que no debe cambiar en un tiempo determinado?									
No			No sé			Sí			
4. Si tiene acceso a un sistema informático que le permita la gestión de sus nomencladores. ¿Lo integraría a su solución?									
6. Una vez integrado su solución al Sistema Nomenclador. ¿Le gustó la solución obtenida?	Sí	No sé	No	Sí	No sé	No	Sí	No sé	No
Me gusta mucho	1	2	6	2	2	6	6	6	6
No me gusta tanto	2	2	3	2	3	3	6	3	6
Me da lo mismo	3	3	3	3	3	3	3	3	3

CÁPITULO 3: ANÁLISIS Y VALORACIÓN DE LOS RESULTADOS

Me disgusta más de lo que me gusta	6	3	6	3	4	4	3	4	4
No me gusta nada	6	6	6	6	4	4	6	4	5
No sé que decir	2	3	6	3	3	3	6	3	4

El índice de satisfacción grupal fluctúa entre + 1 y - 1. Los valores que se encuentran comprendidos entre - 1 y - 0,5 indican insatisfacción; los comprendidos entre - 0,49 y + 0,49 evidencian contradicción y los que caen entre 0,5 y 1 indican que existe satisfacción. El valor obtenido al aplicar la técnica IADOV a la encuesta fue 0.76 y se encuentra entre el intervalo de satisfacción, por lo que se puede concluir que existe satisfacción al integrar los sistemas desarrollados en el Departamento Sistemas de Apoyo a la Salud (SAS) del Centro de Informática Médica (CESIM), al Sistema Nomenclador.

3.4 Análisis económico de la solución

Para verificar el grado de disminución de los gastos en la creación de los sistemas informáticos, se hace uso de una ficha de costo (ver **Anexo 6**) utilizada en la Universidad de las Ciencias Informáticas (UCI). Esta ficha se utilizó en ambos momentos de desarrollo: antes y después de la integración con el Sistema Nomenclador.

Para realizar este análisis solo se tuvo en cuenta el gasto de la mano de obra, por ser donde más influye la solución propuesta. Los roles implicados fueron, *Analista*, *Desarrollador*, *Arquitecto* y *Jefe de proyecto* para cada sistema integrado; así como el tiempo que cada uno estuvo presente en el desarrollo de los sistemas y el salario que recibieron según su categoría docente y científica.

Tabla 16. Gasto de los sistemas antes de la integración y después de la integración

Sistemas	Gasto de los sistemas (CUP)		Disminución en (CUP)	%
	Antes de la integración	Después de la integración		
alas BAP	131615	94267	39148	29,7
RAM	294749	145774	148975	50,5
CM	253937	153971	99966	39,3
CRM	109508	100062	9446	8,6
Total	789809	492274	297535	37,7

Todos los sistemas desarrollados fueron creados en el Centro de Informática Médica (CESIM), de la Facultad 7, con carácter nacional. En la **Tabla 17**. Gasto total de desarrollo antes de la integración y

después de la integración, se muestra la disminución del costo de desarrollo de los sistemas. El dinero ahorrado pudiera ser aplicado en la producción de software a nivel nacional.

Tabla 17. Gasto total de desarrollo antes de la integración y después de la integración

Gasto total de los sistemas sin integración (CUP)	Gasto total de los sistemas con integración (CUP)	Ahorro total
789809	492274	297535

Queda evidenciado cómo el uso del Sistema Nomenclador pudiera ayudar al desarrollo del software desde el punto de vista económico, sin tener que comercializarlo como aplicación; simplemente haciendo uso libre del sistema.

3.5 Valoración de la innovación y aporte práctico de los resultados. Beneficios

Una vez concluida la investigación es necesario evaluar la innovación, aporte práctico y capacidad de generalización de la misma, así como analizar si la hipótesis definida para guiarla fue cumplida satisfactoriamente, la cual plantea: *“si se implementa una aplicación que gestione información poco variable en el tiempo y se integra a las aplicaciones informáticas durante su concepción, entonces se logrará disminuir el tiempo de desarrollo de estas”*. Entre los elementos que se pueden destacar, que permiten concluir que la hipótesis se cumple completamente, se encuentran los siguientes:

1. Se implementó el diseño propuesto, utilizando estándares de implementación para lograr la comunicación con el sistema.
2. Constituye un sistema configurable y flexible, permitiendo al usuario personalizar la información que desee gestionar, a través de la creación y asociación de campos. La estructura del código en su creación podrá ser tan flexible como el usuario desee.
3. Completa y centralizada gestión de los requerimientos de información poco variable en el tiempo para cualquier producto que lo necesite, pues se reutilizan los requerimientos que provee el sistema, trayendo consigo una disminución del tiempo de desarrollo de las aplicaciones al reutilizar el software ya probado y especializado en los procesos antes descritos.
4. Aumento de los niveles de integración entre las diferentes aplicaciones web, evitando que cada uno posea de manera aislada la gestión de la información poco variable en el tiempo.

5. Protección a la información gestionada y compartida, evitando que la misma se encuentre en estados inconsistentes o corruptos; logrando que la información común entre varios sistemas, sea gestionada por un solo usuario.
6. Mejora en el rendimiento de las aplicaciones, debido a que la información poco variable en el tiempo no es gestionada por las mismas.
7. Constituye un valor agregado a los productos desarrollados que deseen usarlo, debido a que además de resolver su negocio, tendrán integrado su propio Sistema de Nomenclador.
8. Los sistemas no tendrán que reprogramarse en caso de que la información cambie en un tiempo determinado, solamente la actualizarían y su sistema quedaría intacto.
9. La solución obtenida no solo se limita a las aplicaciones del ámbito de la salud, sino que puede ser usado en cualquier negocio que gestione información poco variable en el tiempo, implicando por tanto una amplia posibilidad de reutilización y generalización.
10. Concepción de una guía que permite emplear los resultados de la presente investigación por parte de los diseñadores y desarrolladores de sistemas.

Actualmente la propuesta de solución permite gestionar los requerimientos de la información poco variable en el tiempo en un conjunto de aplicaciones desarrolladas por el Centro de Informática Médica (CESIM). A continuación se relacionan las mismas y se hace una breve descripción de los procesos que gestionan:

- *Sistema de Balance y Planificación de Insumos Médicos (alas BAP):* sistema diseñado para gestionar, en una unidad de salud, toda la información relacionada con la planificación y el almacén, lográndose la integración entre ambos procesos. El sistema permite realizar y consultar las planificaciones de las unidades de salud en un período de tiempo planificado, logrando que estas sean lo más exactas posible, según las necesidades de cada unidad de salud.
- *Sistema para el control de recetas médicas:* sistema que permite el control de la asignación de recetas médicas en los diferentes niveles de atención de salud, así como conocer a qué personal médico se le entregó un determinado conjunto de recetas médicas para llevar a cabo la prescripción de los medicamentos.

- *Sistema para el control de Reacciones Adversas a Medicamentos (RAM)*: sistema que permite el control de todas las reacciones adversas documentadas en las unidades de salud cubanas, así como tener en control de todos los casos documentados en Cuba en un periodo de tiempo, permitiendo llevar el control de la población que ha sido trata de por una RAM y qué medicamento fue el sospechoso.
- *Sistema para el consumo de medicamentos*: sistema que permite el control del consumo de medicamentos en Cuba, permitiendo conocer del Formulario Nacional de Medicamentos, cuales han sido los más usados en un periodo de tiempo, de esta manera se pueden realizar comparaciones sobre un medicamento en específico, un grupo poblacional o sencillamente un grupo de edades.

La solución propuesta constituye la base para la gestión de la información poco variable en el tiempo, posibilitando la disminución de los tiempos de desarrollo de los sistemas informáticos que la utilicen: objetivo principal de la presente investigación, que fue cumplido satisfactoriamente.

3.6 Conclusiones parciales

La solución obtenida posee una amplia relevancia en el contexto en el que se desarrolla, debido a que Cuba viene realizando importantes esfuerzos en pos del desarrollo de la informática en todas las actividades que realiza. El desarrollo de estos procesos de informatización de la información poco variable en el tiempo, garantiza una forma homogénea y estandarizada de gestionar la información en todos los ámbitos de la sociedad.

CONCLUSIONES

Con la realización de la presente investigación se han cumplido los objetivos propuestos, obteniéndose las siguientes conclusiones:

- El proceso de integración de los sistemas es un hecho inevitable, elemento que debe ser considerado desde el proceso de conceptualización de un nuevo sistema informático.
- El sistema obtenido tiene un alto nivel de reutilización y generalización, permitiendo su integración con cualquier sistema que cuente en su desarrollo con información nomencada.
- Los resultados obtenidos demuestran disminución en los tiempos de desarrollo de las aplicaciones que se integraron al Sistema Nomenclador, satisfacción por las personas que hicieron uso del mismo, así como una disminución (económica) en los costos de desarrollo.

RECOMENDACIONES

Para lograr un sistema que se adapte mejor a las condiciones de los desarrolladores, el autor recomienda:

- *Permitir la comunicación entre nomencladores creados.* Con esta funcionalidad se garantizará la reutilización de la información común para varios nomencladores, permitiendo que un nomenclador pueda agregar a su estructura información de otro nomenclador, evitando que esta sea repetida en la base de datos.
- *Permitir la restricción de valores en posiciones específicas del código de un grupo.* Esta funcionalidad dará la posibilidad de personalizar más la estructura del código, al poder definir valores fijos o rangos de valores en posiciones específicas.
- *Gestionar las trazas del sistema.* Esta funcionalidad permitirá tener un registro de las operaciones que se realizan en el sistema y de quién las ejecuta, manteniendo así un mejor control de la información perteneciente a los nomencladores.
- Incorporar axiomas al modelo ontológico, utilizando un lenguaje formal que permita de esta forma inferir conocimientos.

REFERENCIAS BIBLIOGRÁFICAS

1. (Álvarez, 2001) **Álvarez, Miguel Angel. 2001.** desarrolloWeb.com. [Online] junio 13, 2001. [Citado: noviembre 25, 2010.] Disponible en: <http://www.desarrolloWeb.com/articulos/449.php>.
2. (Arano, 2005) **Arano, Silvia. 2005.** Los tesauros y las ontologías en la Biblioteconomía y la Documentación. *Hipertext.Net*. [En línea] 2005. [Citado el: 13 de marzo de 2012.] <http://www.upf.edu/hipertextnet/numero-3/tesauros.html#3>.
3. (Barbado, 2011) **Barbado, Dra. Dulce María Calvo. 2011.** Glosario de términos farmacológicos. *Formulario Nacional de Medicamentos*. [En línea] 29 de abril de 2011. [Citado el: 24 de octubre de 2011.] Disponible en: <http://glosario.sld.cu/terminos-farmacologicos/2011/04/29/clasificacion-anatomica-terapeutica-y-quimica-atc/>.
4. (Cabrera, y otros, 2004) **Cabrera, L, Kurt, Christopher y Box, Don. 2004.** An Introduction to the Web Services Architecture and Its Specifications. [En línea] Microsoft Corporation, Octubre de 2004. [Citado el: 14 de noviembre de 2011.] Disponible en: <http://msdn.microsoft.com/en-us/library/ms996441.aspx>.
5. (Chávez, 2007) **Cháves, Michael Arias. 2007.** La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. 2. *Ingeniería de requerimientos: conceptos y características. 2.1 ¿Qué son Requerimientos?* [Online] julio 26, 2007. [Citado: noviembre 28, 2011.] Disponible en: http://www.latindex.ucr.ac.cr/intersedes10/10-art_11.pdf. ISSN 1409-4746.
6. (DASUTeN, 2010) **DASUTeN. 2010.** Sistema para Prestadores de DASUTeN. Concordia : s.n., 2010, págs. 1,2.
7. (Delgado, 2006) **Delgado, Lic. Mercedes Moreira. 2006.** GestioPolis.com. *La organización de la información para la gestión del conocimiento en las empresas*. [En línea] octubre de 2006. [Citado el: 20 de octubre de 2011.] Disponible en: <http://www.gestiopolis1.com/recursos7/Docs/ger/organizacion-de-la-informacion-para-la-gestion-del-conocimiento.htm>.

8. (Díaz, 2009) **Díaz, Dr. Miguel Eusebio Marín. 2009.** Informática en salud 2009. [En línea] febrero de 2009. [Citado el: 20 de octubre de 2011.] Disponible en: <http://informatica2009.sld.cu/Members/marin/nomencladores-medicos-nacionales-para-la-informatizacion-de-la-atencion-medica-en-el-sistema-nacional-de-salud/>.
9. (Diccionario, 2011) **Diccionario. 2011.** El mundo .es. [En línea] 2011. [Citado el: 21 de octubre de 2011.] Disponible en: <http://diccionarios.elmundo.es/diccionarios>.
10. (Fernández, 2009) **Fernández, Andrés. 2009.** infouniversidades. *Divulgación y noticias universitarias*. [En línea] Universidad Nacional de Córdoba, 8 de 1 de 2009. [Citado el: 20 de 10 de 2011.] Disponible en: http://infouniversidades.siu.edu.ar/noticia.php?titulo=nomenclador_c_artografico_para_ciegos&id=273.
11. (Ferrí, 2009) **Ferrí, María Teresa Romá. 2009.** OntoFIS: Tecnología ontológica en el dominio farmacoterapéutico. *OntoFIS: Tecnología ontológica en el dominio farmacoterapéutico*. Alicante : Departamento de Lenguajes y Sistemas Informáticos, Universidad de Alicante, 2009, Capítulo II, Estado de la cuestión, 2.1 Descripción del concepto de `ontología`, pág. 31.
12. (González,2004) **González, Benjamín. 2004.** desarrolloWeb.xom. [En línea] 7 de julio de 2004. [Citado el: 2011 de febrero de 27.] Disponible en: <http://www.desarrolloWeb.com/articulos/1557.php>.
13. (Gruber, 1993) **Gruber, Thomas R. 1993.** *Toward Principles for the Design of Ontologies*. Palo Alto : s.n., 1993. pág. 5. ksl-web.stanford.edu/knowledge-sharing/papers/onto-design.rtf.
14. (ISO27001, 2005) **ISO27001. 2005.** The ISO 27000 Directory. *An Introduction to ISO 27001, ISO 27002.ISO 27008*. [En línea] ISO. Organización Internacional de Estandarización. Sistemas de gestión de seguridad de la información. 2008, octubre de 2005. [Citado el: 2011 de octubre de 1.] Disponible en: <http://www.27000.org/iso-27001.htm>.
15. (Kicillof, y otros, 2004) **Kicillof, Nicolás y Reynoso, Carlos. 2004.** Estilos y Patrones en la estrategia de arquitectura de Microsoft. *Model-View-Controller (MVC)*. [En línea] 1.0, Marzo de 2004. [Citado el: 14 de noviembre

- de 2011.] Disponible en:
<http://carlosreynoso.com.ar/archivos/arquitectura/Estilos.PDF>.
16. (Marco, 2011) **Marco, Bartolomé Sintés. 2011.** mclibre.org. *Páginas Web con PHP*. [Online] noviembre 24, 2011. [Citado: noviembre 28, 2011.] Disponible en:
http://www.mclibre.org/consultar/php/lecciones/php_expresiones_regulares.html.
17. (Noy, y otros, 2005) **Noy, Natalya F. y McGuinness, Deborah L. 2005.** *Ontology Development 101: A Guide to Creating Your First Ontology. What is in an ontology?* [En línea] Stanford University, 19 de septiembre de 2005. [Citado el: 14 de junio de 2012.]
http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html.
18. (OWASP, 2005) **OWASP. 2005.** OWASP, The Open Web Application Security Project. *Una Guía para Construir Aplicaciones y Servicios Web Seguros*. [En línea] julio de 2005. [Citado el: 15 de noviembre 1 de 2011.] Disponible en:
https://www.owasp.org/images/b/b2/OWASP_Development_Guide_2.0.1_Spanish.pdf.
19. (Particulares, 2008) **Particulares, Asociación de Clínicas. 2008.** Superintendencia nacional de aseguramiento en salud. [En línea] marzo de 2008. [Citado el: 20 de 10 de 2011.] Disponible en:
http://www.seps.gob.pe/servicios/nomenclador/nomenclador_presentacion.aspx?opcion=12&seccion=178.
20. (PCMAG, 2010) **PCMAG. 2010.** PCMAG.com. *Enciclopedia*. [En línea] 2010. [Citado el: 15 de noviembre de 2011.]
http://www.pcmag.com/encyclopedia_term/0,2542,t=Web+application&i=54272,00.asp.
21. (Pessano, 2010) **Pessano, Ing. Horacio Paulino. 2010.** D.A.S.Te.N. *Dirección de Acción Social de la Universidad Tecnológica Nacional*. [En línea] 2010. [Citado el: 20 de 10 de 2011.] Disponible en:
<http://www.dasuten.utn.edu.ar/sistema-para-prestadores>.
22. (Possetto, 2010) **Possetto, Ing. Benito Cesar. 2010.** D.A.S.U.Te.N. *Dirección de Acción Social de la Universidad Tecnológica Nacional*. [En línea]

2010. [Citado el: 20 de octubre de 2011.] Disponible en: <http://www.dasuten.utn.edu.ar/sistema-para-prestadores>.
23. (Potencier, 2008) **Potencier, Fabien and Zaninotto, Francois. 2008.** librosWeb.es. *Guía definitiva de Symfony*. 1. 2008, Capítulo 2. Explorando el interior de, pp. 18-425. 2.1. El patrón MVC.
24. (Pressman, 2005) **Pressman, Roger S. 2005.** Ingeniería de Software, un enfoque práctico. *Parte 1*. La Habana : Felix Varela, 2005.
25. (Rodríguez, 2002) **Rodríguez, Alejandro López y Maura, Viviana González. 2002.** Año 8 No 47, Buenos Aires : La técnica de ladov Una aplicación para el estudio de la satisfacción de los alumnos por las clases de educación física, Abril de 2002, Revista Digital - Buenos Aires.
26. (Ronda, 2004) **Ronda. 2004.** Desarrollo de aplicaciones Web multicapas sobre plataforma "Open Source": experiencias de diseño e implementación. [aut. libro] Yoel Ronda Amador. *Tesis para optar por el título de Máster en Informática Aplicada*. Cuba : CUJAE, 2004.
27. (Saladrigas, 2004) **Saladrigas, María Verónica. 2004.** El sistema de clasificación ATC de sustancias, (The Anatomical, Therapeutic, Chemical Classification System). *Panacea@*. [Online] marzo 2004. [Citado: noviembre 28, 2011.] Disponible en: http://medtrad.org/panacea/IndiceGeneral/n15_tribuna-Saladrigas.pdf. Número 15.
28. (Vélez, 2007) **Vélez, Miguel Alejandro Lopera. 2007.** La organización administrativa del deporte y la recreación en 20 instituciones que atienden a población de y en situación de calle del municipio de Medellín. Sistematización de Experiencias. Medellín : s.n., 2007. Universidad de Antioquia.
29. (Wurman, 2001) **Wurman, Richard Saul. 2001.** *Angustia Informativa*. s.l. : Pearson Education, 2001.
30. (W3C, 2010) **W3C, World Wide Web Consortium. 2010.** Guía breve de Servicios Web. [En línea] 6 de mayo de 2010. [Citado el: 10 de octubre de 2011.] Disponible en: <http://www.w3c.es/divulgacion/guiasbreves/ServiciosWeb>.

BIBLIOGRAFÍA

1. **Álvarez, Miguel Angel. 2001.** desarrolloweb.com. [En línea] 13 de junio de 2001. [Citado el: 25 de noviembre de 2010.] Disponible en: <http://www.desarrolloweb.com/articulos/449.php>.
2. **Arano, Silvia. 2005.** Los tesauros y las ontologías en la Biblioteconomía y la Documentación. *Hipertext.Net*. [En línea] 2005. [Citado el: 13 de marzo de 2012.] <http://www.upf.edu/hipertextnet/numero-3/tesauros.html#3.3>.
3. **Barbado, Dra. Dulce María Calvo. 2011.** Glosario de términos farmacológicos. *Formulario Nacional de Medicamentos*. [En línea] 29 de abril de 2011. [Citado el: 24 de octubre de 2011.] Disponible en: <http://glosario.sld.cu/terminos-farmacologicos/2011/04/29/clasificacion-anatomica-terapeutica-y-quimica-atc/>.
4. **Booth, David y Liu, Canyang Kevin. 2007.** W3C Recommendation. *Web Services Description Language (WSDL) Version 2.0 Part 0: Primer*. [En línea] 26 de junio de 2007. [Citado el: 14 de noviembre de 2011.] Disponible en: <http://www.w3.org/TR/wsdl20-primer/#Introduction>.
5. **Booth, David, Haas, Hugo y McCabe, Francis. 2004.** Web Services Architecture. *W3C Working Group Note*. [En línea] World Wide Web Consortium, 11 de febrero de 2004. [Citado el: 14 de noviembre de 2011.] Disponible en: <http://www.w3.org/TR/ws-arch/>.
6. **Brea, Orlando Fabián. 2005.** desarrolloweb.com. *Introducción a los Web Services en PHP*. [En línea] 3 de marzo de 2005. [Citado el: 4 de noviembre de 2011.] Disponible en: <http://www.desarrolloweb.com/articulos/1852.php>.
7. **Cabrera, L, Kurt, Christopher y Box, Don. 2004.** An Introduction to the Web Services Architecture and Its Specifications. [En línea] Microsoft Corporation, Octubre de 2004. [Citado el: 14 de noviembre de 2011.] Disponible en: <http://msdn.microsoft.com/en-us/library/ms996441.aspx>.
8. **Canales Hernández, Anabel, Escobar Martínez, Marisol y Fernández Zapata, Ana Karen. 2012.** *Tamaño de la muestra*. 2012. Estadística Inferencial, Universidad de Veracruzana.
9. **Centelles, Miquel. 2005.** Hipertext.Net. *Taxonomías para la categorización y la organización de la información en sitios web*. [En línea] Universitat de Barcelona, 2005. [Citado el: 17 de enero de 2012.] <http://www.upf.edu/hipertextnet/numero-3/taxonomies.html>. 3.
10. **Cháves, Michael Arias. 2007.** La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. 2. *Ingeniería de requerimientos: conceptos y características*. 2.1 *¿Qué son Requerimientos?* [En línea] 26 de julio de 2007. [Citado el: 28 de noviembre de 2011.] Disponible en: http://www.latindex.ucr.ac.cr/intersedes10/10-art_11.pdf. ISSN 1409-4746.

11. **DASUTeN. 2010.** Sistema para Prestadores de DASUTeN. Concordia : s.n., 2010, págs. 1,2.
12. **Delgado, Lic. Mercedes Moreira. 2006.** GestioPolis.com. *La organización de la información para la gestión del conocimiento en las empresas.* [En línea] octubre de 2006. [Citado el: 20 de octubre de 2011.] Disponible en: <http://www.gestiopolis1.com/recursos7/Docs/ger/organizacion-de-la-informacion-para-la-gestion-del-conocimiento.htm>.
13. **Díaz, Dr. Miguel Eusebio Marín. 2009.** Informática en salud 2009. [En línea] febrero de 2009. [Citado el: 20 de octubre de 2011.] Disponible en: <http://informatica2009.sld.cu/Members/marin/nomencladores-medicos-nacionales-para-la-informatizacion-de-la-atencion-medica-en-el-sistema-nacional-de-salud/>.
14. **Diccionario. 2011.** El mundo .es. [En línea] 2011. [Citado el: 21 de octubre de 2011.] Disponible en: <http://diccionarios.elmundo.es/diccionarios>.
15. EcuRed. *Expresión regular.* [En línea] [Citado el: 28 de noviembre de 2011.] Disponible en: http://www.ecured.cu/index.php/Expresi%C3%B3n_regular.
16. **EL ROL DE LAS ONTOLOGÍAS EN LOS SI. Barchini, Graciela, y otros. 2007.** 14, mayo de 2007, Revista Ingeniería Informática. Facultad de Ciencias Exactas y Tecnologías. Universidad Nacional de Santiago del Estero.
17. *El sistema de clasificación ATC de sustancias.* **Saladrigas, María Verónica. 2004.** 15, 2004, Panacea@, Vol. V, pág. 58 y 59. Disponible en: http://medtrad.org/panacea/IndiceGeneral/n15_tribuna-Saladrigas.pdf.
18. **Fernández, Andrés. 2009.** infouniversidades. *Divulgación y noticias universitarias.* [En línea] Universidad Nacional de Córdoba, 8 de 1 de 2009. [Citado el: 20 de 10 de 2011.] Disponible en: http://infouniversidades.siu.edu.ar/noticia.php?titulo=nomenclador_cartografico_para_ciegos&id=273.
19. **Ferrí, María Teresa Romá. 2009.** OntoFIS: Tecnología ontológica en el dominio farmacoterapeutico. *OntoFIS: Tecnología ontológica en el dominio farmacoterapeutico.* Alicante : Departamento de Lenguajes y Sistemas Informáticos, Universidad de Alicante, 2009, Capítulo II, Estado de la cuestión, 2.1 Descripción del concepto de `ontología`, pág. 31.
20. **González, Benjamín. 2004.** desarrolloweb.xom. [En línea] 7 de julio de 2004. [Citado el: 2011 de febrero de 27.] Disponible en: <http://www.desarrolloweb.com/articulos/1557.php>.
21. **Gruber, Thomas R. 1993.** *Toward Principles for the Design of Ontologies.* Palo Alto : s.n., 1993. pág. 5. ksl-web.stanford.edu/knowledge-sharing/papers/onto-design.rtf.

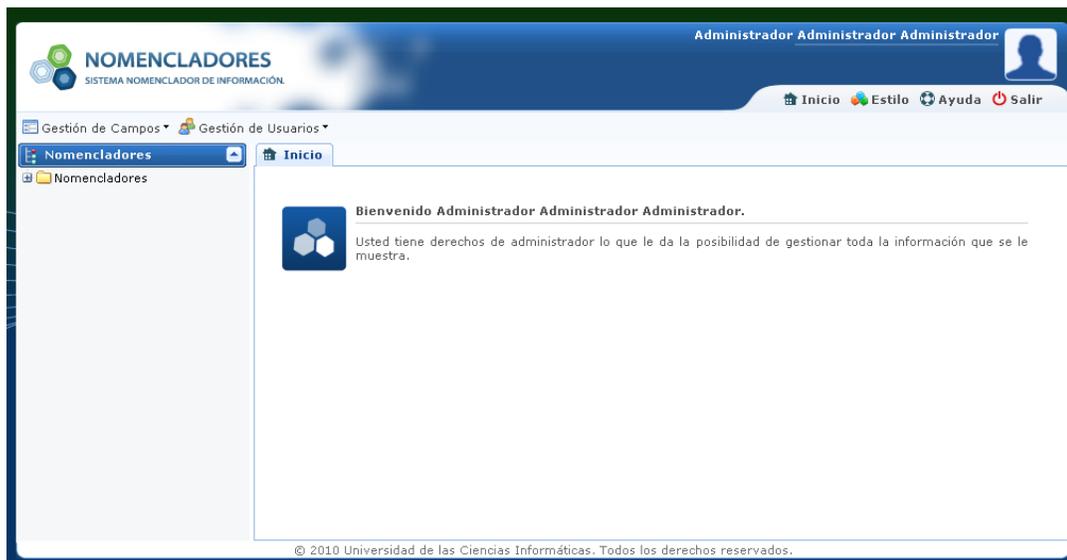
22. *Instrumentos de presentación del conocimiento: Teauros versus ontologías*. **García Jiménez, Antonio**. 2004. 7, España : s.n., 2004, Anales de documentación, pág. 4. Red de Revistas Científicas de América Latina y el Caribe, España y Portugal, Universidad Autónoma del Estado de México. ISSN 1697-7904.
23. **ISO27001**. 2005. The ISO 27000 Directory. *An Introduction to ISO 27001, ISO 27002.ISO 27008*. [En línea] ISO. Organización Internacional de Estandarización. Sistemas de gestión de seguridad de la información. 2008, octubre de 2005. [Citado el: 2011 de octubre de 1.] <http://www.27000.org/iso-27001.htm>.
24. **Kicillof, Nicolás y Reynoso, Carlos**. 2004. Estilos y Patrones en la estrategia de arquitectura de Microsoft. *Model-View-Controller (MVC)*. [En línea] 1.0, Marzo de 2004. [Citado el: 14 de noviembre de 2011.] Disponible en: <http://carlosreynoso.com.ar/archivos/arquitectura/Estilos.PDF>.
25. **Lorente Casafont, Mercè**. 2005. Hipertext.net. *Ontología sobre economía y recuperación de información*. [En línea] Universitat Pompeu Fabra, 2005. [Citado el: 15 de junio de 2012.] http://www.upf.edu/hipertextnet/numero-3/ontologia_ri.html#3. 3.
26. **Marañón, Gonzalo Álvarez**. 2003. Institut Seguritat Internet. *Boletín de Criptonomicón*. [En línea] 5 de febrero de 2003. [Citado el: 24 de octubre de 2011.] Disponible en: <http://www.instisec.com/publico/verarticulo.asp?id=70>.
27. **Marco, Bartolomé Sintés**. 2011. mclibre.org. *Páginas web con PHP*. [En línea] 24 de noviembre de 2011. [Citado el: 28 de noviembre de 2011.] Disponible en: http://www.mclibre.org/consultar/php/lecciones/php_expresiones_regulares.html.
28. **Microsoft, Sun**. 2002. Building Web Services. *Building Web Services*. [En línea] 2002. [Citado el: 14 de noviembre de 2011.] Disponible en: <http://www.sun.com>.
29. **Noy, Natalya F. y McGuinness, Deborah L**. 2005. Ontology Development 101: A Guide to Creating Your First Ontology. *What is in an ontology?* [En línea] Stanford University, 19 de septiembre de 2005. [Citado el: 14 de junio de 2012.] http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html.
30. *Ontología para representar el conocimiento de*. **Medina Velandia, Lucy Nohemy y Escobar Escobar, Álvaro**. 2011. [ed.] Universidad Sergio Arboleda. 1, Medellín : s.n., 09 de febrero de 2011, Revista Avances en Sistemas e Informática, Vol. 8, pág. 8. ISSN 1657-7663.
31. *Ontología: un atécnica de represetación de conocimiento*. **Velásquez Pérez, Torcoroma, Puenetes Velásquez, Andres Mauricio y Guzmán Luna, Jaime Alberto**. 2011. 2, Medellín : s.n., julio de 2011, Universidad Sergio Arboleda, Vol. 8, pág. 6. Universidad Sergio Arboleda.

32. *Ontologies, Taxonomies and Thesaurus in Systems*. **Currás, Emilia**. 2012. 2, mayo-agosto de 2012, Ciencias de la Información, Vol. 43, pág. 73.
33. **OWASP**. 2005. OWASP, The Open Web Application Security Project. *Una Guía para Construir Aplicaciones y Servicios Web Seguros*. [En línea] julio de 2005. [Citado el: 15 de noviembre de 2011.] https://www.owasp.org/images/b/b2/OWASP_Development_Guide_2.0.1_Spanish.pdf .
34. **OWASP**. 2010. OWASP. *The Open WEB Application Security Project*. [En línea] 2010. [Citado el: 15 de noviembre de 2011.] https://www.owasp.org/index.php/Main_Page.
35. **Particulares, Asociación de Clínicas**. 2008. Superintendencia nacional de aseguramiento en salud. [En línea] marzo de 2008. [Citado el: 20 de 10 de 2011.] Disponible en: http://www.seps.gob.pe/servicios/nomenclador/nomenclador_presentacion.aspx?opcion=12&seccion=178.
36. **PCMAG**. 2010. PCMAG.com. *Enciclopedia*. [En línea] 2010. [Citado el: 15 de noviembre de 2011.] http://www.pcmag.com/encyclopedia_term/0,2542,t=Web+application&i=54272,00.asp.
37. **Pessano, Ing. Horacio Paulino**. 2010. D.A.S.Te.N. *Dirección de Acción Social de la Universidad Tecnológica Nacional*. [En línea] 2010. [Citado el: 20 de 10 de 2011.] Disponible en: <http://www.dasuten.utn.edu.ar/sistema-para-prestadores>.
38. **Possetto, Ing. Benito Cesar**. 2010. D.A.S.U.Te.N. *Dirección de Acción Social de la Universidad Tecnológica Nacional*. [En línea] 2010. [Citado el: 20 de octubre de 2011.] Disponible en: <http://www.dasuten.utn.edu.ar/sistema-para-prestadores>.
39. **Potencier, Fabien y Zaninotto, Francois**. 2008. librosweb.es. *Guía definitiva de Symfony*. 1. 2008, Capítulo 2. Explorando el interior de, págs. 18-425. 2.1. El patrón MVC.
40. **Pressman, Roger S**. 2005. Ingeniería de Software, un enfoque práctico. *Parte 1*. La Habana : Felix Varela, 2005.
41. **Real Academia Española**. Real Academia Española. *DICCIONARIO DE LA LENGUA ESPAÑOLA - Vigésima segunda edición*. [En línea] [Citado el: 20 de octubre de 2011.] Disponible en: http://buscon.rae.es/draeI/SrvltGUIBusUsual?TIPO_BUS=3&LEMA=nomenclador.
42. **Rodríguez, Alejandro López y Maura, Viviana González**. 2002. Año 8 No 47, Buenos Aires : La técnica de ladov Una aplicación para el estudio de la satisfacción de los alumnos por las clases de educación física, Abril de 2002, Revista Digital - Buenos Aires.
43. **Ronda**. 2004. Desarrollo de aplicaciones Web multicapas sobre plataforma "Open Source": experiencias de diseño e implementación. [aut. libro] Yoel Ronda Amador. *Tesis para optar por el título de Máster en Informática Aplicada*. Cuba : CUJAE, 2004.

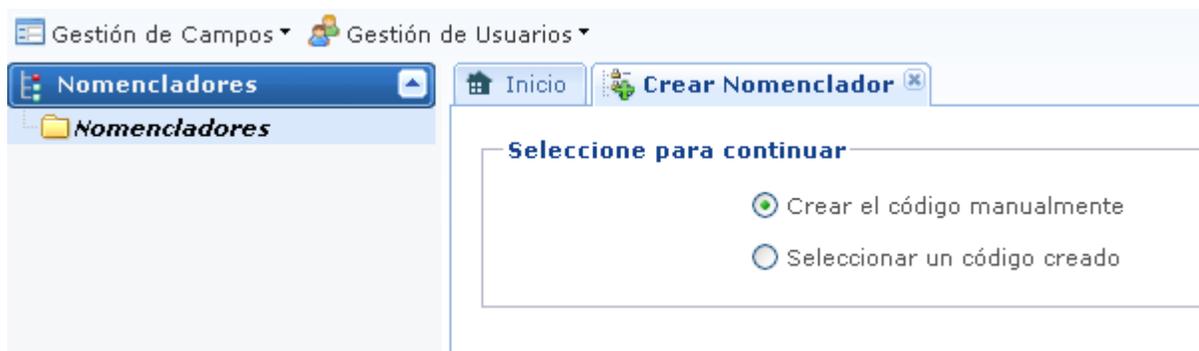
44. **Saladrigas, María Verónica. 2004.** El sistema de clasificación ATC de sustancias, (The Anatomical, Therapeutic, Chemical Classification System). *Panacea@*. [En línea] marzo de 2004. [Citado el: 28 de noviembre de 2011.] Disponible en: http://medtrad.org/panacea/IndiceGeneral/n15_tribuna-Saladrigas.pdf. Número 15.
45. **TAMAÑO DE LA MUESTRA. Mateu, Enric; Casal, Jordi. 2003.** 1, 2003, Revista epidemiológica medicina prevalente. Universitat Autònoma de Barcelona.
46. **Tello, Adolfo Lozano. 2002.** *MÉTRICA DE IDONEIDAD DE ONTOLOGÍAS*. Departamento de Informática, Escuela Politécnica de Cáceres, UNIVERSIDAD DE EXTREMADURA. Extremadura : UNIVERSIDAD DE EXTREMADURA, 2002. pág. 236, Tesis doctoral.
47. **Vélez, Miguel Alejandro Lopera. 2007.** La organización administrativa del deporte y la recreación en 20 instituciones que atienden a población de y en situación de calle del municipio de Medellín. Sistematización de Experiencias. Medellín : s.n., 2007. Universidad de Antioquia.
48. **W3C, World Wide Web Consortium. 2010.** Guía breve de Servicios Web. [En línea] 6 de mayo de 2010. [Citado el: 10 de octubre de 2011.] Disponible en: <http://www.w3c.es/divulgacion/guiasbreves/ServiciosWeb>.
49. **Wurman, Richard Saul. 2001.** *Angustia Informativa*. s.l. : Pearson Education, 2001.

ANEXOS

Anexo 1: Interfaz principal del sistema



Anexo 2: Creación del código del nomenclador



2.1 Creación del código del nomenclador

 Inicio  **Crear Nomenclador** 

Definición del separador

- 

Caracteres por posición

Caracter 1: Solo Letra Mayúscula 

Caracter 2: Solo Letra Minúscula 

Caracter 3: Solo Número 

- Solo Letra Mayúscula
- Solo Letra Minúscula
- Solo Número
- Solo Número/Letra

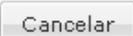
Anexo 3: Creación del nomenclador

 Inicio  **Nuevo Nomenclador** 

Datos del Nomenclador

Nombre:  Código:  Descripción: 

Grupo:

Anexo 4: Encuesta aplicada a los desarrolladores y futuros clientes del Sistema Nomenclador

Encuesta sobre integración con el Sistema Nomenclador

Dpto. productivo: _____

Rol: _____

El desarrollo de todo sistema informático necesita de un conjunto de información que no varía en un tiempo determinado, más conocida como nomencladores. Responda según su criterio, las preguntas siguientes:

1. ¿Tiene conocimiento de sistemas que realicen de forma flexible la gestión de la información nomenclada adaptable a cualquier sistema?

Sí_____ No_____ No sé_____

*En caso de ser **negativa** su respuesta argumente.*

2. ¿Al desarrollar un sistema informático tiene en cuenta la estructura de la información que no debe cambiar en un tiempo determinado?

Sí_____ No_____ No sé_____

*En caso de ser **negativa** su respuesta argumente.*

3. Al desarrollar una aplicación informática que cuente con varios nomencladores y Ud. debe desarrollarlos todos, ¿Qué le causaría? Marque con una X las que Ud. cree que se ponen de manifiesto.

 Aumento del tiempo de desarrollo del sistema informático. Descentralización de la información que pueda ser común entre varios módulos. Desgaste físico del personal que desarrolla las aplicaciones. Cambio total o parcial en el sistema al cambiar la estructura de la información nomenclada.

Otras: _____

4. Si tiene acceso a un sistema informático que le permita la gestión de sus nomencladores ¿Lo integraría a su solución?

Sí_____ No_____ No sé_____

Si la respuesta es positiva responda el inciso a y si es negativa responda el inciso b.

a) Marque con una X las ventajas que Ud. considera que le permita la integración.

- _____ Disminuir el tiempo de desarrollo del sistema informático.
 _____ Permitir que la información común entre varios módulos esté centralizada.
 _____ El sistema integrado es poco vulnerable al cambio de la información incorporada en el Sistema Nomenclador. (Sí cambia la información no se afecta el sistema)

Otros: _____

b) Marque con una X las posibles causas por las cuales Ud. considere que no es necesaria la integración.

- _____ Se aumenta el tiempo de desarrollo del sistema informático porque los desarrolladores no cuentan con suficiente conocimiento para realizar la integración con el Sistema Nomenclador.
 _____ Considero que es innecesario el uso de un sistema externo para la creación y configuración de los nomencladores del sistema.
 _____ Me siento más cómodo desarrollando yo mismo los requerimientos de la información nomenclada aunque se atrase el desarrollo del sistema.

Otros: _____

5. ¿Ud. ha integrado el Sistema Nomenclador alguna solución ya desarrollada?

Sí_____ No_____

*En caso de ser **positiva** su respuesta, marque con una X las posibles causas.*

- _____ Fue factible, se realizaron cambios en el sistema que permitieron mejorar el funcionamiento y gestión de la información nomenclada.
 _____ El Sistema Nomenclador cuenta con una detallada descripción de los servicios web que permitieron una rápida integración.

Otras: _____

*En caso de ser **negativa** su respuesta, marque con una X las posibles causas*

- Ha sido engorrosa la integración con el Sistema Nomenclador.
 Se realizaron demasiados cambios en el sistema informático, atrasando demasiado el desarrollo.
 El equipo de desarrollo no tenía conocimiento del trabajo con servicios web.
 La descripción de los servicios web no estaba completa.
 Se tenía que desarrollar el sistema desde cero.

Otras: _____

6. Una vez integrado su solución al Sistema Nomenclador. ¿Le gustó la solución obtenida?

- | | | | |
|--------------------------|-------------------|--------------------------|------------------------------------|
| <input type="checkbox"/> | Me gusta mucho | <input type="checkbox"/> | Me disgusta más de lo que me gusta |
| <input type="checkbox"/> | No me gusta tanto | <input type="checkbox"/> | No me gusta nada |
| <input type="checkbox"/> | Me da lo mismo | <input type="checkbox"/> | No sé que decir |

Marque con una X las posibles causas que apoyan su selección.

- Disminuyó el tiempo de desarrollo.
 La información en el Sistema Nomenclador permitirá una fácil modificación, eliminación o cambio en la estructura organizativa sin que se vea afectado el sistema informático.
 Mi sistema será más flexible a los posibles cambios sin tener que cambiar la implementación.
 El grupo de desarrollo cuenta con suficiente experiencia en integración de aplicaciones a través de servicios web.
 No me hace falta saber cómo se creó el Sistema Nomenclador, siempre y cuando me resuelva el problema de gestionar la información nomenclada.

Otros: _____

Anexo 5: Tamaño muestral

Para determinar el tamaño muestral del estudio se utilizó la siguiente fórmula:

Estimar una proporción:

- a. El nivel de confianza o seguridad (1-a). El nivel de confianza prefijado da lugar a un coeficiente (Z_a). Para una seguridad del 95% = 1.96, para una seguridad del 99% = 2.58.

- b. La precisión que se desea para el estudio.
- c. Una idea del valor aproximado del parámetro que se quiere medir (en este caso una proporción). Esta idea se puede obtener revisando la literatura, por estudios pilotos previos. En caso de no tener dicha información se utiliza el valor $p = 0.5$ (50%).

Seguridad = 95%; Precisión = 5%: Proporción esperada = se asume que puede ser próxima al 5%; si no se tuviera ninguna idea de dicha proporción se utilizaría el valor $p = 0,5$ (50%) que maximiza el tamaño muestral:

$$n = \frac{Z_a^2 * p * q}{d^2}$$

donde:

- $Z_a^2 = 1.962$ (ya que la seguridad es del 95%)
- $p =$ proporción esperada (en este caso $80\% = 0.8$)
- $q = 1 - p$ (en este caso $1 - 0.8 = 0.2$)
- $d =$ precisión (en este caso es un 20%)

$$x = \frac{1.962^2 * 0,8 * 0,2}{0,2^2} = 15,39$$

Anexo 6: Ficha de costo

Ficha de costo del sistema Reacciones Adversas a Medicamentos antes de la integración con el Sistema Nomenclador.

Cargo	Cantidad	Mes	Salario	Importe
Analista	2	3	800	2400
Desarrollador	2	8	1500	12000
Arquitecto	1	8	750	6000
Jefe Proyecto	1	10	836	8360
	Tasa	Base		
Vacaciones	9.09%	28760	2614	31374
Impuesto Fuerza Trabajo	25%	31374	7843	39217
Seguridad Social	14%	31374	4392	35766
Total				135117