

Universidad de las Ciencias Informáticas

Facultad 3



Título: “Solución de software para la gestión de configuración y carga inicial del Sistema Integral de Gestión de Entidades Cedrux.”

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas

Autor: Yannier Estévez Dieppa

Tutor: Ing. René Lazo Ochoa

Ciudad de la Habana, junio de 2009

DEDICATORIA

A Lety, mi madre, por levantar el Sol con tus propias manos cuando las noches se hacían eternas.

A Raquel, mi abuela, por opacar mi tristeza con tu alegría.

A Estela, mi hermana, por abrazarme con tus alas y ser mi ejemplo.

A Javier, mi padre, por tu apoyo incondicional.

A Amaris, por tu presencia en mi vida.

AGRADECIMIENTOS

A René, mi tutor, por guiar mis pasos.

A Irina, por los buenos tiempos tardíos.

A todos mis amigos, en especial a Guille, Andy y Albertón, por compartir estos años.

A todos aquellos que aportaron tiempo y conocimiento.

RESUMEN

La Universidad de las Ciencias Informáticas (UCI) tiene entre sus misiones fundamentales la producción de software y servicios informáticos, con el objetivo de la informatización de la sociedad cubana y como una importante rama de la economía. Para ello en la UCI se desarrollan programas nacionales de informatización y proyectos de exportación.

Uno de los programas de mayor prioridad, por la importancia en el orden económico y de control, es el programa para la creación de un Sistema Integral de Gestión de Entidades. A este producto se le ha dado como nombre *Cedrux*. Este es un sistema complejo, que a su vez, agrupa varios subsistemas de acuerdo a las áreas o procesos básicos de la gestión de cualquier empresa o entidad. Para el correcto y óptimo funcionamiento de esta aplicación es necesario contar con información inicial que permita el ajuste de la aplicación a cada organismo, así como la información mínima para comenzar su utilización.

El presente trabajo de diploma responde a esta última necesidad. Brinda una solución de flujos de trabajo¹ para guiar y ordenar la carga inicial de datos del sistema *Cedrux*. Se realiza un estudio de distintos frameworks² que permitan la modelación de flujos de trabajo, se define la corriente a seguir y se abordan los pasos necesarios para implementar la solución propuesta. Por último se realiza una evaluación de la solución.

¹ Flujo de trabajo: Conocido internacionalmente como workflow. Es la automatización de un proceso de negocio, total o parcial, en la cual documentos, información o tareas son pasadas de un participante a otro para efectos de su procesamiento, de acuerdo a un conjunto de reglas establecidas. (Workflow Management Coalition)

² Framework: Estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado.

ÍNDICE

Índice de contenidos:

INTRODUCCIÓN	1
CAPÍTULO 1	5
FUNDAMENTACIÓN TEÓRICA	5
INTRODUCCIÓN.....	5
1.1 HISTORIA Y DEFINICIÓN DE FLUJO DE TRABAJO.....	5
1.2 BASE TEÓRICA MATEMÁTICA	5
1.2.1 <i>Redes de Petri (RdP)</i>	6
1.2.2 <i>Definición Red de Petri Ordinaria (RPO)</i>	7
1.3 PRINCIPALES ORGANIZACIONES QUE RIGEN EL DESARROLLO TEÓRICO.....	9
1.3.1 <i>Coalición de Administración de Flujos de trabajo (WfMC)</i>	9
1.3.2 <i>Grupo de Administración de Objetos (OMG)</i>	10
1.3.3 <i>Instituto Nacional para Estándares y Tecnologías (NIST)</i>	10
1.4 CLASIFICACIÓN DE LOS DIFERENTES TIPOS DE FLUJOS DE TRABAJO.....	11
1.4.1 <i>Flujo de trabajo de producción</i>	12
1.4.2 <i>Flujo de trabajo de colaboración</i>	12
1.4.3 <i>Flujo de trabajo de administración</i>	12
1.4.4 <i>Flujo de trabajo Ad-Hoc</i>	13
1.5 MODELO DE REFERENCIA	13
1.6 INTEGRACIÓN CON OTRAS DISCIPLINAS.....	16
1.6.1 <i>Administración de Procesos de Negocio (BPM)</i>	16
1.6.2 <i>Arquitectura Orientada a Servicios (SOA)</i>	17
1.7 PRINCIPALES HERRAMIENTAS PARA LA GESTIÓN DE FLUJOS DE TRABAJO	17
1.8 CONCEPTUALIZACIÓN Y DEFINICIÓN DE LA CORRIENTE A SEGUIR	19
1.9 COMO SE CONCIBE DESARROLLAR	19
CONCLUSIONES	20
CAPÍTULO 2	21

DESCRIPCIÓN DE LA SOLUCIÓN.....	21
INTRODUCCIÓN	21
2.1 REQUISITOS	21
2.1.1 Funcionales	21
2.2.1 No funcionales.....	23
2.2 PROTOTIPO DE INTERFAZ.....	26
2.4.1 Descripción del prototipo de interfaz.....	27
2.5 PATRONES UTILIZADOS.....	28
2.6 DIAGRAMA DE COMPONENTES.....	29
2.7 DISEÑO DE CLASES	30
2.8 MODELO DE DATOS	33
2.9 CASOS DE PRUEBA.....	35
CONCLUSIONES	40
CAPÍTULO 3	41
EVALUACIÓN DE LA SOLUCIÓN.....	41
INTRODUCCIÓN	41
3.1 MÉTRICAS QUE EVALÚAN EL DISEÑO.....	41
3.1.1 Métricas de software	41
3.1.2 Métricas orientadas a objetos.....	42
3.1.3 Métricas orientadas a clases	42
3.1.4 Métricas aplicadas a la solución propuesta	43
3.2 MATRIZ DE INFERENCIA DE INDICADORES DE CALIDAD.....	57
CONCLUSIONES	58
CONCLUSIONES GENERALES	59
RECOMENDACIONES.....	60
BIBLIOGRAFÍA.....	61
GLOSARIO DE TÉRMINOS	64
ANEXOS	66
ANEXO 1: INSTRUMENTO DE MEDICIÓN DE LA MÉTRICA DE TAMAÑO OPERACIONAL DE CLASES (TOC).	66
ANEXO 2: INSTRUMENTO DE MEDICIÓN DE LA MÉTRICA DE RELACIONES ENTRE CLASES (RC).	67

ANEXO 3: INSTRUMENTO DE MEDICIÓN DE LA MÉTRICA DE PROFUNDIDAD DE HERENCIA (PH).....	68
ANEXO 4: INSTRUMENTO DE MEDICIÓN DE LA MÉTRICA DE NÚMERO DE DESCENDIENTES (ND).	69
ANEXO 5: INSTRUMENTO DE MEDICIÓN DE LA MÉTRICA DE NÚMERO DE OPERACIONES REDEFINIDAS (NOR).	70
ANEXO 6: DIRECCIONES DE EJECUCIÓN Y CHEQUEO DE LAS TAREAS.	71

Índice de tablas:

Tabla 1. Comparación de herramientas para la gestión de flujos de trabajo.	18
Tabla 2. Descripción del requisito R1.....	22
Tabla 3. Descripción del requisito R2.....	22
Tabla 4. Descripción del requisito R3.....	22
Tabla 5. Descripción del requisito R4.....	23
Tabla 6. Descripción del requisito R5.....	23
Tabla 7. Descripción de la clase controladora PizarraController.	31
Tabla 8. Descripción de la clase del negocio DatWorkflowModel.....	32
Tabla 9. Descripción de la clase del negocio DatChequeoModel.....	32
Tabla 10. Descripción de la clase entidad DatTarea.	32
Tabla 11. Descripción de la clase entidad DatFlujo.....	33
Tabla 12. Descripción de la clase entidad DatEstadotarea.	33
Tabla 13. Descripción de la clase entidad DatChequeo.	33
Tabla 14. Descripción de la clase entidad DatConfiguracion.....	33
Tabla 15. DCP verificar configuración de la carga inicial.....	36
Tabla 16. DCP permitir acceder al portal.	36
Tabla 17. DCP cargar los flujos y las tareas de la carga inicial.	37
Tabla 18. DCP verificar las precondiciones y postcondiciones de cada tarea.	38
Tabla 19. DCP permitir ejecutar solamente las tareas que tengan cumplidas sus precondiciones.....	39
Tabla 20. Tamaño de clases (TOC).	44
Tabla 21. Respuesta para una clase (RC).	44
Tabla 22. Árbol de profundidad de herencia (PH).	45
Tabla 23. Número de descendientes (ND).	45
Tabla 24. Número de operaciones redefinidas para una sub-clase (NOR).....	45

Tabla 25. Matriz de inferencia de indicadores de calidad.....	57
Tabla 26. Rango de valores para la evaluación técnica de los atributos de calidad evaluados por cada métrica.....	58
Tabla 27. Rango de valores para la evaluación técnica de los atributos de calidad (Responsabilidad, Complejidad de implementación y Reutilización) relacionados con la métrica TOC.....	66
Tabla 28. Resultados de la evaluación de la métrica TOC y su influencia en los atributos de calidad (Responsabilidad, Complejidad de Implementación y Reutilización).....	66
Tabla 29. Rango de valores de para la evaluación técnica de los atributos de calidad (Acoplamiento, Complejidad de Mantenimiento, Reutilización y Cantidad de Pruebas) relacionados con la métrica RC.....	68
Tabla 30. Resultados de la evaluación de la métrica RC y su influencia en los atributos de calidad (Acoplamiento, Complejidad de Mantenimiento, Reutilización y Cantidad de Pruebas).....	68
Tabla 31. Resultados de la evaluación de la métrica PH.....	68
Tabla 32. Rango de valores de para la evaluación técnica de los atributos de calidad (Reutilización, Abstracción del diseño, Nivel de Cohesión y Cantidad de Pruebas) relacionados con la métrica ND.....	69
Tabla 33. Resultados de la evaluación de la métrica ND y su influencia en los atributos de calidad (Reutilización, Abstracción del diseño, Nivel de Cohesión y Cantidad de Pruebas).....	69
Tabla 34. Rango de valores de para la evaluación técnica de los atributos de calidad (Abstracción del diseño, Cantidad de pruebas, Complejidad de Mantenimiento) relacionados con la métrica NOR.....	70
Tabla 35. Resultados de la evaluación de la métrica NOR y su influencia en los atributos de calidad (Abstracción del diseño, Cantidad de pruebas, Complejidad de Mantenimiento).....	71

Índice de figuras:

Figura 1. Lugares, transiciones y señales en una Red de Petri. (Cervantes Canales, 2005).....	7
Figura 2. Representación lineal RdP. (Bergmann).....	8
Figura 3. Representación Iterativa RdP. (Bergmann).....	8
Figura 4. Representación Y-Dividir RdP. (Bergmann).....	8
Figura 5. Representación Y-Unir RdP. (Bergmann).....	8
Figura 6. Representación O-Dividir RdP. (Bergmann).....	9
Figura 7. Representación O-Unir RdP. (Bergmann).....	9
Figura 8. Modelo de referencia de flujos de trabajo. (Hollingsworth).....	15
Figura 9. Concepción de desarrollo.....	20
Figura 10. Prototipo de interfaz.....	27
Figura 11. Diagrama de componentes.....	29

Figura 12. Diagrama de clases.	30
Figura 13. Diagrama de clases del paquete de Dominio.	31
Figura 14. Modelo de datos	34
Figura 15. Representación de los resultados obtenidos en el instrumento agrupados en los intervalos definidos.	46
Figura 16. Representación en % de los resultados obtenidos en el instrumento agrupados en los intervalos definidos.	47
Figura 17. Representación de la incidencia de los resultados de la evaluación de la métrica TOC en el atributo Responsabilidad.....	47
Figura 18. Representación de la incidencia de los resultados de la evaluación de la métrica TOC en el atributo Complejidad de Implementación.	48
Figura 19. Representación de la incidencia de los resultados de la evaluación de la métrica TOC en el atributo Reutilización.....	48
Figura 20. Representación de los resultados de la evaluación de la métrica RC agrupados por la tendencia de los valores.....	49
Figura 21. Representación en % de los resultados obtenidos en el instrumento agrupados en los intervalos definidos.	49
Figura 22. Representación de la incidencia de los resultados de la evaluación de la métrica RC en el atributo Acoplamiento.	50
Figura 23. Representación de la incidencia de los resultados de la evaluación de la métrica RC en el atributo Complejidad de Mantenimiento.	50
Figura 24. Representación de la incidencia de los resultados de la evaluación de la métrica RC en el atributo Cantidad de Pruebas.....	51
Figura 25. Representación de la incidencia de los resultados de la evaluación de la métrica RC en el atributo Reutilización.....	51
Figura 26. Representación en % de los resultados obtenidos en el instrumento agrupados por nivel.	52
Figura 27. Representación de la incidencia de los resultados de la evaluación de la métrica ND en el atributo Reutilización.....	53
Figura 28. Representación de la incidencia de los resultados de la evaluación de la métrica ND en el atributo Abstracción de la clase base.....	53
Figura 29. Representación de la incidencia de los resultados de la evaluación de la métrica ND en el atributo Cohesión de la Jerarquía de clases.	54
Figura 30. Representación de la incidencia de los resultados de la evaluación de la métrica ND en el atributo Cantidad de Pruebas.....	54
Figura 31. Representación en % de los resultados obtenidos en el instrumento agrupados en los valores existentes.....	55

Figura 32. Representación de la incidencia de los resultados de la evaluación de la métrica NOR en el atributo Complejidad del Mantenimiento.	56
Figura 33. Representación de la incidencia de los resultados de la evaluación de la métrica NOR en el atributo Cantidad de Pruebas.....	56
Figura 34. Representación de la incidencia de los resultados de la evaluación de la métrica NOR en el atributo Violación de la Abstracción representada por la superclase.....	56
Figura 35. Gráfica de los resultados obtenidos de los atributos de calidad evaluados en las métricas.....	58
Figura 36. Gráfica de los resultados de la evaluación de la métrica TC y su influencia en los atributos de calidad (Responsabilidad, Complejidad de Implementación y Reutilización).....	67
Figura 37. Gráfica de los resultados de la evaluación de la métrica PH.	68
Figura 38. Representación de los resultados obtenidos al aplicar los instrumentos que evalúan la métrica ND.	70
Figura 39. Representación de los resultados obtenidos al aplicar los instrumentos que evalúan la métrica NOR.....	71

INTRODUCCIÓN

La tendencia en el mundo actual para la informatización de diferentes empresas y organismos es su gestión a través de programas de informatización o modernización, teniendo como base fundamental los procesos³ que en ellas se desarrollen.

Esto provoca que el desarrollo de sistemas que implementan procesos empresariales o la gestión de diferentes procesos, tenga una fuerte presencia y sean cada vez mayores en número. Actualmente existen varias soluciones informáticas a nivel mundial para disminuir en gran medida los problemas empresariales. Estas soluciones son conocidas como Sistemas Integrales de Gestión.

La función de un Sistema Integral de Gestión está centrada principalmente en integrar y manejar la mayoría de los negocios que se asocian a las operaciones de producción y distribución que realizan las empresas caracterizadas por la producción de bienes o servicios.

La tecnología de automatización de flujos de trabajo se ha erigido, en los últimos años, como una herramienta de gran relevancia y eficiencia para llevar a cabo la necesaria coordinación de los elementos que intervienen en los procesos: usuarios, actividades, recursos y reglas de actuación. Los sistemas basados en flujos de trabajo⁴ permiten una automatización integral del entorno de los procesos, aportando el dinamismo necesario para gestionar adecuadamente la complejidad y heterogeneidad de tales elementos. (González, 2006)

Existen un gran número de productos basados en flujos de trabajo, los cuales generalmente permiten llevar las tareas a todos los responsables y que las aplicaciones se puedan conectar definiendo un conjunto de reglas y recursos necesarios para completar el proceso de negocio.

³ Proceso: Es un conjunto de uno o más procedimientos o actividades directamente ligadas, que colectivamente realizan un objetivo del negocio normalmente dentro del contexto de una estructura organizacional que define roles funcionales y relaciones entre los mismos. (Workflow Management Coalition)

⁴ Sistema basado en flujos de trabajo: Es un sistema que define, crea y gestiona automáticamente la ejecución de modelos workflow mediante el uso de uno o más motores workflow que se encargan de: interpretar la definición de procesos (mapa workflow), interactuar con los agentes y, cuando se requiera, invocar el uso de los sistemas de información implicados en el trabajo. (Workflow Management Coalition)

Una aplicación de flujos de trabajo automatiza toda la secuencia de procedimientos utilizados para que un proceso sea ejecutado. Se incluye a esto, el seguimiento de cada una de las etapas del proceso y las herramientas para su gestión.

La implantación de estos sistemas basados en flujos de trabajo arroja beneficios dependiendo de los procesos de negocio que estén involucrados, como son: mejora del control de los procesos, ahorro de tiempo, integración de los procesos empresariales y optimización del flujo de información.

Algunos de los sistemas basados en flujos de trabajo más reconocidos y robustos del mercado son *LiquidOffice*⁵, *IBM Lotus Workflow*⁶ y *openEDMS*⁷.

Una característica importante de estos sistemas es que es posible llevar a cabo una correcta integración con sistemas actuales, como bases de datos, mensajería y sistemas de gestión, entre otros.

La Universidad de las Ciencias Informáticas (UCI) tiene entre sus misiones fundamentales la producción de software y servicios informáticos, con el objetivo de la informatización de la sociedad cubana y como una importante rama de la economía. Para ello, se desarrollan programas nacionales de informatización y proyectos de exportación.

Uno de los programas de mayor prioridad, por la importancia en el orden económico y de control, es el programa para la creación de un Sistema Integral de Gestión de Entidades, a este producto se le ha dado como nombre *Cedrux*. Este es un sistema complejo que a su vez agrupa varios subsistemas de acuerdo a las áreas o procesos básicos de la gestión de cualquier empresa o entidad. *Cedrux* en su primera versión consta con 14 subsistemas, más de 109 componentes de software, flujos de integración externos entre subsistemas en el orden de 117 y flujos de integración internos en el orden de 164. La cantidad de pasos

⁵ LiquidOffice: Solución basada en Web, para la creación, implementación y administración automática de los procesos de enrutamiento, seguimiento y aprobación de formas electrónicas.

⁶ IBM Lotus Workflow: Es una herramienta de desarrollo de flujos de trabajo de aplicaciones, independiente y fácil de usar, que hereda las capacidades del flujo de trabajo nativo del software IBM Lotus Domino y acelera la creación y el desarrollo de aplicaciones orientadas a flujos de trabajo.

⁷ openEDMS: Herramienta que permite gestionar gran cantidad de documentos y hacer más eficiente la creación de contenido de negocio, almacenar, revisar, distribuir y compartir la información con mayor agilidad.

a ejecutar para realizar la configuración y carga inicial del sistema es de 82 por cada entidad, por lo que se visualiza la dimensión del problema. Para el correcto y óptimo funcionamiento de esta aplicación es necesario contar con información inicial que permita el ajuste de la aplicación a cada organismo así como la información mínima para comenzar su utilización.

Cuando se accede al sistema *Cedrux* por primera vez el usuario debe cargar los datos iniciales necesarios para que pueda funcionar correctamente. Si esto no se hace de una forma ordenada se puede convertir en un proceso muy complejo y difícil, afectando la usabilidad del sistema y la interpretación de los clientes, pudiendo influir, por la complejidad del proceso, en un indicador que afecte la aceptación del producto. Por lo antes planteado debe existir una guía que le indique qué datos debe entrar al sistema en cada instante del proceso.

Dada estas características, es necesario que la solución se gestione en base a flujos de trabajo. De ese modo, se pueden utilizar al máximo los beneficios que brindan los sistemas de este tipo, logrando una mayor flexibilidad, reutilización y adaptación a los cambios de negocio durante el proceso de desarrollo, permitiendo también adicionar nuevas funcionalidades o características al proceso de configuración, con costes menores que si se desarrolla de forma tradicional.

Por tanto, es necesaria la implementación de un componente basado en flujos de trabajo que se encargue de la gestión del proceso de la carga inicial del sistema *Cedrux* y sirva de guía para el usuario.

El componente debe gestionar un gran número de tareas las cuales pertenecen a un proceso de ejecución que tiene un orden lógico.

Por las situaciones expuestas anteriormente este trabajo de diploma se propone darle solución al siguiente **problema**: ¿Cómo organizar de manera flexible, adaptativa a los cambios de negocio, con alta usabilidad y comprensión para el usuario el proceso de configuración y carga inicial del sistema de gestión integral *Cedrux*?

El **objeto de estudio** de este trabajo queda definido como solución de software de Sistemas de Gestión de Entidades.

El **objetivo general** del trabajo consiste en solucionar la gestión de la configuración y carga inicial del Sistema de Gestión Integral de Entidades *Cedrux*.

Del objetivo general se define para este trabajo como **campo de acción** la solución de software de la configuración y carga inicial de los Sistemas de Gestión de Entidades

Para darle cumplimiento al objetivo general definido para el trabajo quedan definidos los siguientes **objetivos específicos**:

- Realizar informe referencial del estado del arte de la teoría de flujos de trabajo, y modelos de referencia de soluciones genéricas basadas en flujos de trabajo.
- Realizar levantamiento de los requisitos de la solución de configuración y carga inicial del sistema de gestión de entidades *Cedrux*.
- Diseñar solución de software para la gestión de la configuración y carga inicial del sistema de gestión de entidades *Cedrux*.
- Desarrollar la solución para la gestión de la configuración y carga inicial del sistema de gestión de entidades *Cedrux*.
- Realizar valoración de la solución propuesta.

Con la culminación de este trabajo se espera obtener un componente basado en flujos de trabajo que permita solucionar la gestión de la configuración y carga inicial del Sistema de Gestión Integral de Entidades *Cedrux*.

El trabajo consta de tres capítulos. En el capítulo uno se aborda la fundamentación teórica, se incluyen herramientas, metodologías y técnicas usadas en la actualidad referidas al objeto de estudio. En el capítulo dos se describe la solución, contiene requisitos, diseño de clases, artefactos de implementación y los casos de prueba. Por último y en el capítulo tres, se evalúa y valora la solución.

FUNDAMENTACIÓN TEÓRICA

Introducción

En este capítulo se realiza un estudio sobre la teoría de flujos de trabajo desde su base teórica matemática hasta las tendencias y corrientes actuales del tema. Se aborda la integración del flujo de trabajo con otras disciplinas y las principales herramientas o frameworks para la gestión de los mismos. Se realiza una conceptualización de la corriente a seguir y se explica como se concibe desarrollar la solución.

1.1 Historia y definición de flujo de trabajo

El concepto de flujo de trabajo en la historia moderna se remonta a 1912 cuando F. Taylor y a H. Gantt, juntos iniciaron el estudio de la organización racional de trabajo, referido al ámbito de la manufactura. (Saffirio)

Este concepto fue evolucionando y aplicándose a diferentes ramas, fundamentalmente en la informática, donde se conceptualiza que:

Un flujo de trabajo es un conjunto de uno o más procedimientos o actividades directamente ligadas, que colectivamente realizan un objetivo del negocio normalmente dentro del contexto de una estructura organizacional que define roles funcionales y relaciones entre los mismos. (Workflow Management Coalition)

Los modelos de flujo de trabajo tienen una fuerte base matemática para su implementación, por lo que es necesario para un mejor entendimiento, conocer la misma.

1.2 Base teórica matemática

Los flujos de trabajo tienen una fuerte base teórica matemática, centradas en las Redes de Petri (RdP).

1.2.1 Redes de Petri (RdP)

Las RdP fueron introducidas en la literatura en la tesis doctoral de Carl Adam Petri como una herramienta para simular las propiedades dinámicas de sistemas complejos mediante modelos gráficos de procesos concurrentes. Desde entonces su estudio y desarrollo han tenido un auge realmente vigoroso debido fundamentalmente a las numerosas aplicaciones que se les han encontrado, las cuales incluyen diversas áreas del conocimiento y de la técnica. Se ha demostrado que las RdP son un instrumento adecuado para la representación y análisis de ciertos sistemas, ya que estas tienen la habilidad de representar y analizar de una forma fácil sincronización y concurrencia, donde varios procesos que evolucionan simultáneamente son parcialmente independientes. (Cervantes Canales, 2005)

Las RdP son una herramienta muy adecuada para el modelado y análisis de los procesos de negocio de las organizaciones. Por una parte se pueden utilizar como lenguaje de diseño para la especificación de complejos flujos de trabajo, y por otra, la teoría de RdP proporciona una potente herramienta de análisis para verificar la corrección de los procedimientos de flujos de trabajo. (The application of Petri nets to workflow management., 1998)

Las RdP se describen como una herramienta de naturaleza gráfica para el diseño y análisis de sistemas dinámicos de eventos discretos. Una RdP se representa gráficamente por un grafo dirigido bipartito. Los dos tipos de nodos, lugares y transiciones (ver Figura 1. Lugares, transiciones y señales en una Red de Petri.) representan, las variables que definen el estado del sistema (lugares) y sus transformadores (transiciones). Los lugares se representan por círculos, las transiciones por barras y el marcado M se representa por una distribución en los lugares denominados 'marcas'. Una marca se representa gráficamente por un punto en el interior del círculo que define el lugar que lo contiene. Los lugares y transiciones se conectan por arcos dirigidos. Un arco dirigido de un lugar P_i a una transición T_j define un lugar de entrada de la transición. Múltiples entradas a una transición son indicadas por múltiples arcos desde el lugar de entrada a la transición. Un lugar de salida es indicado por un arco desde la transición al lugar. Análogamente, múltiples salidas son representadas por múltiples arcos. (Cervantes Canales, 2005)

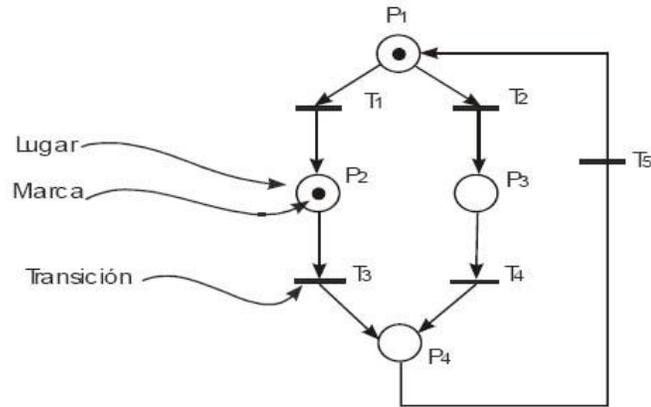


Figura 1. Lugares, transiciones y señales en una Red de Petri. (Cervantes Canales, 2005)

Existen varias definiciones de RdP, pero la que interesa es la Red de Petri Ordinaria (RPO).

1.2.2 Definición Red de Petri Ordinaria (RPO)

Una RPO N , es un cuádruplo $N = \langle P, T, Pre, Post \rangle$, donde (Cervantes Canales, 2005):

$P = \{p_1, p_2, \dots, p_m\}$ es un conjunto finito y no vacío de lugares;

$T = \{t_1, t_2, \dots, t_n\}$ es un conjunto finito y no vacío de transiciones;

$P \cap T = \emptyset$ y $P \cup T = \emptyset$;

$Pre: P \times T \rightarrow \{0, 1\}$ es el conjunto de lugares de entrada a T ;

$Post: T \times P \rightarrow \{0, 1\}$ es el conjunto de lugares de salida de T .

Una RPO marcada es un par $N_m = (N, M_0)$ en el cual N es una red de Petri Ordinaria

y M_0 es el marcado inicial.

Las RdP fueron propuestas para el modelado de flujos de trabajo por van derAalst (van derAalst, 1996), porque ellas proveen una *semántica formal a pesar de la naturaleza gráfica* y una *existencia de*

abundantes técnicas de análisis. Desde la figura 2 hasta la 6 se muestra como el flujo de trabajo primitivo⁸ del modelo de referencia (ver epígrafe 1.5 Modelo de referencia) puede ser expresado usando RdP (Bergmann)

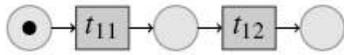


Figura 2. Representación lineal RdP. (Bergmann)

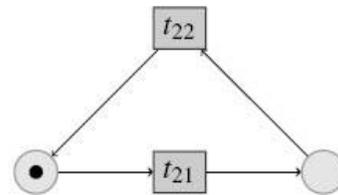


Figura 3. Representación Iterativa RdP. (Bergmann)

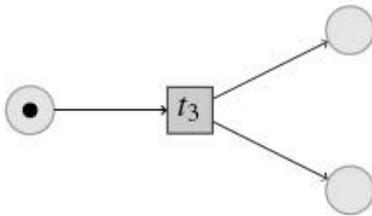


Figura 4. Representación Y-Dividir RdP. (Bergmann)

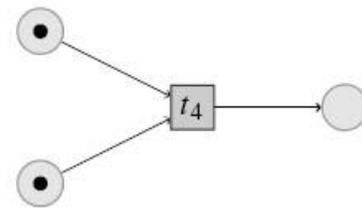


Figura 5. Representación Y-Unir RdP. (Bergmann)

⁸ Flujo de trabajo primitivo: Es aquel que no es definido en función de otro flujo de trabajo, de forma que es un flujo de trabajo que debe ser ejecutado por un agente (humano o no), sin necesidad de más definición.

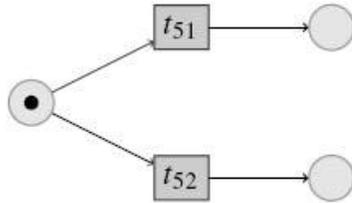


Figura 6. Representación O-Dividir RdP. (Bergmann)

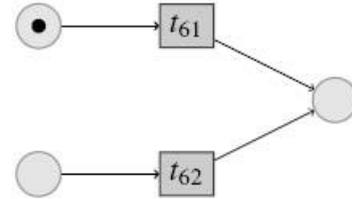


Figura 7. Representación O-Unir RdP. (Bergmann)

1.3 Principales organizaciones que rigen el desarrollo teórico

Actualmente existen muchas organizaciones que han realizado aportes en el área del flujo de trabajo, entre ellas se destacan; la *Coalición de Administración de Flujos de trabajo (WfMC)*, el *Grupo de Administración de Objetos (OMG)* y el *Instituto Nacional para Estándares y Tecnologías (NIST)*.

Estas organizaciones han realizado muchos esfuerzos por crear definiciones y estándares que ayuden a un mejor manejo, comprensión y gestión del flujo de trabajo.

1.3.1 Coalición de Administración de Flujos de trabajo (WfMC)

Fundada en 1993, WfMC es una organización global de desarrolladores, consultores, analistas, así como universidades y grupos que promueven y centran investigaciones en torno al flujo de trabajo y la Administración de Procesos de Negocio (BPM). La WfMC crea y contribuye con el proceso relacionado con estándares, educa al mercado en cuestiones y temas relacionados en esta área y es la única organización que se concentra puramente en los procesos. La WfMC creó Wf-XML⁹ y XPDL¹⁰, la definición

⁹ Wf-XML: Ofrece un estándar para un motor BPM para invocar un proceso en otro motor BPM y para esperar su completamiento.

¹⁰ XPDL: Es un formato de archivo basado en XML que puede ser usado para intercambiar modelos de procesos de negocio entre distintas herramientas.

de procesos líder usada hoy en más de 80 soluciones conocidas para almacenar e intercambiar modelos de procesos. (Workflow Management Coalition)

La WfMC cuenta con más de 300 organizaciones miembros en todo el mundo, lo que representa todas las facetas de workflow, desde los proveedores hasta los usuarios, y desde los académicos hasta los consultores. (Workflow Management Coalition)

1.3.2 Grupo de Administración de Objetos (OMG)

Fundada en 1989, el OMG es un consorcio dedicado al cuidado y el establecimiento de diversos estándares de tecnologías orientadas a objetos, tales como UML¹¹, XMI¹² y CORBA¹³. Es una organización que promueve el uso de tecnología orientada a objetos mediante guías y especificaciones para las mismas. El grupo está formado por más de 800 compañías y organizaciones de software tales como *Hewlett-Packard (HP)*, *IBM*, *Sun Microsystems* y *Apple Computer*. (OMG)

El OMG es el creador de una tecnología de rápida evolución llamada Workflow Management (WfM), explotada por una variedad de industrias. Su principal característica es la automatización de los procesos entre humanos y máquinas basados en actividades, en particular aquellos que impliquen la interacción con aplicaciones y herramientas de las tecnologías de la información (TI). (OMG)

1.3.3 Instituto Nacional para Estándares y Tecnologías (NIST)

Fundado en 1901, el NIST es una agencia federal de regulación dentro del Departamento de Comercio de los Estados Unidos. La misión del NIST es promover la innovación y la competitividad industrial de los Estados Unidos mediante la mejora de la ciencia, los estándares, y la tecnología en formas que optimicen la seguridad económica y la calidad de vida. (National Institute of Standards and Technology.)

¹¹ UML: *Unified Modeling Language* es un lenguaje de modelado de sistemas de software respaldado por el OMG.

¹² XMI: Es una especificación para el intercambio de diagramas, también conocido como XML de Intercambio de Metadatos.

¹³ CORBA: *Common Object Request Broker Architecture* es un estándar que establece una plataforma de desarrollo de sistemas distribuidos facilitando la invocación de métodos remotos bajo un paradigma orientado a objetos, definido y controlado por el OMG.

Investigadores del NIST han completado una versión inicial de una herramienta para el desarrollo internacional de estándares llamada Process Specification Language (PSL).

Ha surgido con el proyecto PSL de NIST el proyecto Process Interchange Format (PIF). El núcleo del PIF (PIF CORE) y sus extensiones han sido incorporados dentro del núcleo (PSL CORE) y las extensiones del PSL. Crítico en la Reingeniería de Procesos de Negocio o en la Integración Empresarial es la habilidad de compartir y vincular modelos de procesos. La meta del proyecto PIF es soportar el intercambio de modelos de procesos de negocio a través de diferentes formatos y esquemas.

El proyecto persigue esta meta desarrollando PIF (incluyendo un lenguaje de traducción común que sirve como puente entre las representaciones de procesos), traductores locales entre PIF y las representaciones de procesos locales, y un mecanismo que permite extender PIF para acomodar diferentes necesidades expresivas en una forma modular.

1.4 Clasificación de los diferentes tipos de flujos de trabajo

Existe una gran variedad de productos catalogados como flujos de trabajo y han sido mucho los intentos por clasificarlos (Carlsen, 1997)

Para muchas personas el área de flujos de trabajo es complicada en cuanto a su entendimiento, por este motivo se ha dividido en diferentes categorías los diferentes tipos de aplicaciones de esta índole.

La variada existencia de procesos de negocio en la actualidad ha llevado a que los diferentes tipos de flujos de trabajo sean clasificados según el valor del proceso a manejar y su repetición. Los procesos tienen alto valor si le representa a una empresa grandes ahorros o si aporta grandes beneficios para la misma. Se puede decir que un proceso es repetitivo cuando cada instancia sigue ciertas reglas y patrones similares para toda instancia del proceso, por otra parte no es repetitivo si cada instancia es única.

Tomando como base las consideraciones anteriores, se pueden clasificar los flujos de trabajo en cuatro grupos fundamentales: (Ramos, y otros)

- De producción.
- De colaboración.
- Administrativo.
- Ad-Hoc.

1.4.1 Flujo de trabajo de producción

Este tipo de flujo de trabajo con frecuencia también es llamado de transacciones, ya que en este caso, la transacción en una base de datos es considerada el elemento clave de cada proceso.

El flujo de trabajo de producción es actualmente considerado el más amplio del mercado, en general automatizan procesos de negocio que tienden a ser repetitivos, con gran manejo de datos y bien estructurados.

Se utiliza para transacciones de emisiones críticas de alto valor y volumen. Estos sistemas son muy robustos porque manejan un número enorme de transacciones. Los procesos que usan este modelo necesitan estar muy estructurados. Involucran procesos de negocios predecibles y repetitivos. (Ramos, et al.)

El orden y la coordinación de tareas puede ser automatizada, sin embargo esto último no es una tarea sencilla debido a la complejidad del proceso de información y el acceso a múltiples sistemas de información para recuperar datos necesarios para la toma de decisiones. (Ramos, et al.)

1.4.2 Flujo de trabajo de colaboración

Las aplicaciones de flujos de trabajo que resuelven procesos de negocios donde participa gente para lograr una meta común, son llamadas flujos de trabajo de colaboración. Los mismos estructuran o semi-estructuran procesos de negocios donde participan personas, con el objetivo de lograr una meta común.

Típicamente involucran documentos, los cuales son los contenedores de la información. Se sigue la ruta de estos paso a paso, además de las acciones que se toman sobre ellos. Los documentos son la clave, y por lo tanto es esencial para la solución de flujos de trabajo mantener la integridad de dichos documentos. (Canseco, y otros)

1.4.3 Flujo de trabajo de administración

El flujo de trabajo administrativo, como lo dice su nombre, es aquel que involucra procesos de administración en una empresa tales como órdenes de compra, reportes de ventas, etc. Estos flujos de trabajo se emplean cuando existe una gran cantidad de procesos administrativos dentro de la empresa y es necesaria la distribución de soluciones a diferentes usuarios.

Una solución de flujo de trabajo administrativo difiere para cada organización, y los cambios son frecuentes. Por esto, la posibilidad de poder hacer cambios de diseño es muy importante. (Canseco, et al.)

1.4.4 Flujo de trabajo Ad-Hoc

Realizan procesos de oficina donde no hay un patrón para distribuir la información entre la gente. Las tareas de este tipo de flujo de trabajo involucran coordinación, colaboración y co-decisión, es decir, el orden y la coordinación no están automatizados pero son controlados por personas. Este modelo involucra pequeños equipos de profesionales y soportan actividades de corto plazo que requieren una solución rápida. (Ramos, et al.)

1.5 Modelo de referencia

En los últimos años han surgido varias transformaciones importantes en cuanto a tecnologías web, redes de alta complejidad y grupos de trabajo que se interconectan mediante redes internas. Estas transformaciones que influyen con los procesos y actividades han llevado a las empresas a que adopten sistemas basados en flujos de trabajo para implantar modelos mucho más cooperativos y que se basen en el trabajo en equipo para responder de una manera mucho más ágil a los cambio del mercado y enfrentar de una forma eficiente la competencia, aprovechando el conocimiento adquirido en el transcurso del tiempo.

Con todas estas transformaciones nace un problema: poder definir estándares para desarrollar estos sistemas. Por esta razón surge un modelo de referencia de flujos de trabajo para identificar características similares entre estos sistemas, proporcionando así un marco general para la construcción de los mismos y poder garantizar interoperabilidad con otras aplicaciones relacionadas y entre los mismos sistemas basados en workflow.

Se toma el modelo de referencia (ver Figura 8. Modelo de referencia de flujos de trabajo.) descrito por David Hollingsworth. (Hollingsworth)

El modelo de referencia fue desarrollado por la WfMC (ver epígrafe 1.3.1 Coalición de Administración de Flujos de trabajo (WfMC)) para tener una estructura genérica en el desarrollo de aplicaciones de workflows, es decir, un estándar. (Canseco, y otros)

Todos los sistemas basados en flujos de trabajo contienen componentes genéricos que interactúan de forma definida. Para mantener la interoperabilidad entre los diversos productos se definen un conjunto de interfaces y formatos para el intercambio de datos entre dichos componentes. (R. de Soto, et al., 2006)

Los componentes que considera el modelo de referencia (ver Figura 8. Modelo de referencia de flujos de trabajo.) son los siguientes:

El *Motor de Flujos de trabajo* de manera general es un software encargado de administrar y ejecutar procesos.

Los *Servicios de Ejecución de Flujos de trabajo* interpretan y controlan los proceso o las secuencias de actividades, también pueden invocar aplicaciones que sean necesarias. Están integrados por uno o más motores de flujos de trabajo.

La *Interfaz de Programación de Aplicaciones de Flujos de trabajo* (WAPI) es un conjunto de interfaces de programación de aplicaciones (APIs) y funciones de intercambio soportadas por el servicio de ejecución de flujos de trabajo. Permiten la interacción del servicio de ejecución con otros recursos y aplicaciones. (R. de Soto, et al., 2006)

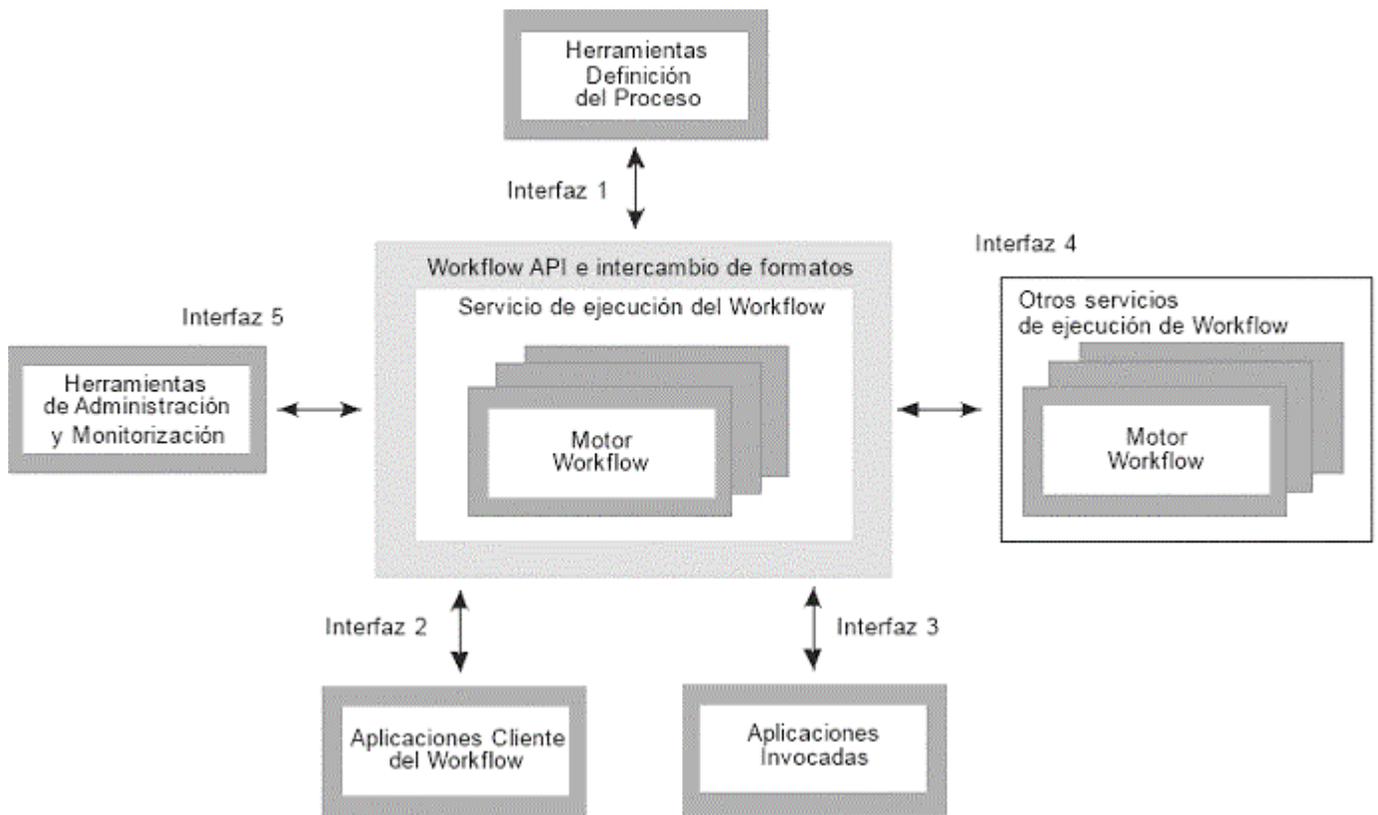


Figura 8. Modelo de referencia de flujos de trabajo. (Hollingsworth)

El modelo presenta interfaces que juegan un rol fundamental en la comprensión del mismo.

- *Interfaz 1: Definición de procesos*

Los analistas se encargan de definir los procesos con las actividades que serán ejecutadas, el orden de las mismas y las condiciones para que pueda comenzar la ejecución de estas. Obteniendo una definición del procesos que podrá ser interpretada por el motor de flujos de trabajo. Esta interfaz se encarga del intercambio de información entre el componente que permite la definición del proceso y el servicio de ejecución de flujo de trabajo.

- *Interfaz 2: Aplicaciones clientes del flujo de trabajo*

Conjunto de APIs que permiten al motor de flujos de trabajo brindar servicios que sean previamente solicitados por las aplicaciones clientes y así poder controlar la progresión de procesos y actividades.

- *Interfaz 3: Aplicaciones invocadas*

Define un conjunto de APIs para que el motor de flujos de trabajo pueda invocar aplicaciones, las cuales pueden ser locales al motor (residentes en la misma plataforma), o estar en otra plataforma dentro de la red. En este caso la definición del proceso debe contener información necesaria para poder encontrar la aplicación a invocar.

- *Interfaz 4: Servicios de ejecución*

Brinda la posibilidad de ejecutar el flujo de trabajo. Es posible que exista interrelación entre diferentes motores cuando el flujo de trabajo es distribuido, los cuales pueden controlar distintas etapas del flujo.

- *Interfaz 5: Administración y monitorización*

Permite una monitorización de la ejecución del flujo de trabajo, mostrando el estado del mismo.

La integración de los sistemas basados en flujos de trabajo con otras disciplinas ha evolucionado durante los últimos años. En los siguientes epígrafes se hace referencia a ella.

1.6 Integración con otras disciplinas

Actualmente existen disciplinas que han surgido después del flujo de trabajo, pero que se integran perfectamente y hacen uso del mismo de manera tal que han alcanzado resultados relevantes. Entre estas disciplina se destacan dos, *Administración de Procesos de Negocio (BPM)* y *Arquitectura Orientada a Servicios (SOA)*.

1.6.1 Administración de Procesos de Negocio (BPM)

BPM se entiende como una metodología empresarial cuyo objetivo es mejorar la eficiencia a través de la gestión sistemática de los procesos de negocio, que se deben modelar, automatizar, integrar, monitorizar y optimizar de forma continua.

Es un conjunto de técnicas, actividades y tareas, bajo un enfoque metodológico o metodología, con el fin de gestionar los procesos de negocio.

Se ha venido empleando el término BPM también para ir reemplazando, desde un punto de vista, el término flujo de trabajo, que está más asociado a tecnologías de los 90. Esto también está dado, porque los flujos de trabajo han ido evolucionando e incorporando nuevas funcionalidades. Se puede concebir BPM como la evolución de los flujos de trabajo, ya que además de definir tareas y darle seguimiento,

usando herramientas basadas en BPM, se puede definir y solucionar el ciclo completo de una administración basada en procesos, desde el diseño del proceso hasta la explotación de la información basada en el mismo. BPM permite integrar aplicaciones que convivan en la empresa y lograr que el proceso trabaje como la empresa lo necesite, integrando las reglas de negocio con el mismo proceso además de toda la lógica de las funcionalidades que requiera la misma.

1.6.2 Arquitectura Orientada a Servicios (SOA)

Básicamente SOA es un tipo de arquitectura de software¹⁴ que se centra en la utilización de servicios para dar soporte a los requisitos del negocio. Un servicio es un componente disponible y reusable tanto dentro como fuera de la red de la empresa.

SOA es una plataforma para software de flujos de trabajo. La mayoría de las soluciones BPM pueden consumir servicios web para ejecutar procesos automáticos. Desde otra perspectiva, un proceso administrado por un motor de flujos de trabajo puede ser visto como un servicio web asíncrono por otra aplicación. Estas funcionalidades pueden ser activadas usando estándares como Wf-XML. Por tanto, una organización que implemente SOA, optimizará considerablemente su administración de procesos mediante una solución de flujos de trabajo.

1.7 Principales herramientas para la gestión de flujos de trabajo

En este epígrafe se realiza una caracterización y comparación entre algunas herramientas para la gestión de flujos de trabajo de licencia libre.

Existen varias herramientas para la gestión de flujos de trabajo, todas de licencia libre. Debido a la gran cantidad de proyectos de este tipo se considera a *OpenWFE*, *Enhydra Shark* y *Bonita* porque son los que más están presentes en foros de debate, blogs y comunidades; cada una de ellas, se analiza con más detalle a continuación (ver **¡Error! No se encuentra el origen de la referencia.**).

En la Tabla 1 se evalúa las potencialidades de algunas de las características que deben estar presentes en los sistemas de flujos de trabajo, donde se establecen 4 escalas: Mal (M), Regular (R), Bien (B) y Excelente (E).

¹⁴ Arquitectura de software: Básicamente indica la estructura, funcionamiento e interacción entre las partes del software.

Características	OpenWFE	Enhydra Shark	Bonita
Versión	1.7.1	2.0	3.0
Licencia	BSD	LPGL	LPGL
Soporte	R	B	B
Funcionalidad	R	B	B
Madurez	B	R	E
Diseño gráfico de procesos	R	B	E
Proactivo	R	B	E
Documentación	M	B	E

Tabla 1. Comparación de herramientas para la gestión de flujos de trabajo.

OpenWFE

Es un Sistema de Administración de Flujos de Trabajo (WfMS) creado en el año 2001 y registrado el 29 de Mayo del 2002, está disponible bajo una licencia BSD¹⁵. OpenWFE utiliza a Droflo, que es una herramienta basada en la web y permite el diseño gráfico de procesos. El modelo de los procesos se almacena en formato XPDL.

Enhydra Shark

Es un motor de flujos de trabajo¹⁶ y utiliza el lenguaje de definición de procesos especificado por la WfMC, basado en XML, como formato de definición de flujos de trabajo, sin extensiones propietarias.

Bonita

Es un sistema de flujos de trabajo que ofrece muchos rasgos innovadores como definir actividades que pueden comenzar con anticipación, además ofrece una infraestructura de conocimientos que les permite a los usuarios ser notificados de cualquier evento que ocurra durante la ejecución en un proceso dado, o activación automática del código de usuario según un ciclo de vida de actividad definido. Incorpora también los rasgos de los flujos de trabajo tradicionales como son la definición dinámica de usuarios y roles, la ejecución secuencial de actividades también está incluida en Bonita.

¹⁵ BSD: Licencia de software libre resultado del proyecto llamado *Berkeley Software Distribution* en la universidad del mismo nombre en California en la década de los 70s. La *Free Software Foundation* la considera compatible con la GPL.

¹⁶ Motor de flujos de trabajo: Software encargado de administrar y ejecutar procesos. Algunos motores de workflow pueden interactuar entre ellos.

1.8 Conceptualización y definición de la corriente a seguir

El flujo de trabajo para la implementación del componente puede ser clasificado, según sus características, como flujo de trabajo de producción (ver epígrafe 1.4.1 Flujo de trabajo de producción) ya que las transacciones serán guiadas por precondiciones existentes en una base de datos, además es un proceso que será repetitivo y manejará gran número de datos bien estructurados, relacionados a cada tarea que se encuentre en ejecución en un instante del proceso.

Se decide guiar la solución por el modelo de referencia de flujos de trabajo adoptado por la Coalición de Administración de Flujos de Trabajo (ver epígrafe 1.3.1 Coalición de Administración de Flujos de trabajo (WfMC)) y descrito en el epígrafe 1.5 Modelo de referencia, el cual intenta reunir las características comunes para cualquier sistema de administración de flujos de trabajo. Debido que el objetivo de la Coalición de Administración de Flujos de Trabajo (WfMC) es definir estándares y guías globales para el desarrollo de sistemas de administración de flujos de trabajo, su documentación es de carácter genérico y no describe en detalle ningún sistema en particular, sino que brinda un marco general para la construcción de los mismos.

1.9 Como se concibe desarrollar

Para que la solución sea flexible, se adapte a los cambios de negocio y su usabilidad y comprensión para el usuario sea alta, además de brindarle beneficios al proceso de configuración y carga inicial del sistema de gestión integral *Cedrux* se concibe desarrollar de la siguiente forma:

La pizarra será la encargada de brindarle al usuario la posibilidad de interactuar con la solución, brindándole una alta comprensión del proceso que se esté ejecutando. El usuario podrá conocer en que estado se encuentra el flujo en cada instante del mismo y ejecutar las tareas, que dada las condiciones, estén disponibles (ver Figura 9. Concepción de desarrollo.).

Un motor de flujos de trabajo será el núcleo de la solución, se encargará de la ejecución de cada uno de los flujos e interactuará con la pizarra. También existirá una comunicación con el acceso a datos, permitiendo la estructuración de los flujos para cada entidad. Un chequeo asociado al motor de flujos de trabajo se encargará de manejar las precondiciones y postcondiciones de cada tarea, interactuando con el acceso a datos y los servicios brindados por cada uno de los subsistemas implicados (ver Figura 9. Concepción de desarrollo.).

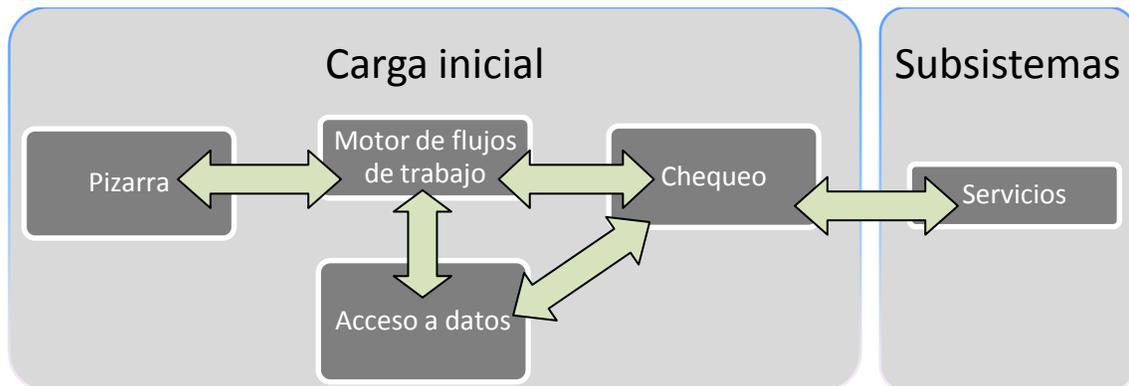


Figura 9. Concepción de desarrollo.

Conclusiones

Se pudo contemplar el surgimiento y la fuerte base matemática de los flujos de trabajo haciendo énfasis en las Redes de Petri y su definición. Se caracterizaron las principales organizaciones que rigen el desarrollo teórico de los flujos de trabajo. Se abordaron los diferentes grupos de clasificación de flujos de trabajo, permitiendo enmarcar la solución en uno de ellos. Se desglosaron cada una de las partes del modelo de referencia a seguir, así como la integración de otras disciplinas con los flujos de trabajo. Por último se realizó una conceptualización y definición de la corriente a seguir durante el desarrollo, de la solución y se concibió una forma de desarrollar la misma.

DESCRIPCIÓN DE LA SOLUCIÓN

Introducción

En este capítulo se describe la solución teniendo en cuenta los requisitos funcionales y no funcionales, incluye una muestra del prototipo interfaz y el diseño de clases. Además se genera el diagrama de clases y artefactos de implementación tales como el diagrama de componentes y el modelo de datos, por último se abordan los casos de prueba establecidos que permiten demostrar al cliente la reacción que corresponderá por parte de la solución luego de realizar alguna acción sobre la misma.

2.1 Requisitos

Los requisitos son características que los sistemas deben tener para cubrir las necesidades que lo motivan. Son útiles, ya que denotan una condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo. Los requisitos funcionales, no funcionales y de interfaz se abordan a continuación.

2.1.1 Funcionales

- R1. Verificar configuración de la carga inicial.
- R2. Permitir acceder al portal.
- R3. Cargar los flujos y las tareas de la carga inicial.
- R4. Verificar las precondiciones y postcondiciones de cada tarea.
- R5. Permitir ejecutar solamente las tareas que tengan cumplida sus precondiciones.

La descripción de cada uno de los requisitos funcionales mencionados anteriormente permite un mejor entendimiento de los mismos.

Descripción de los requisitos funcionales

Requisito R1

Precondiciones	Precondiciones	Pre-requisito
	El usuario debe seleccionar una entidad.	
Descripción	Verifica que la configuración de la carga inicial haya sido completada para la entidad que intenta acceder al portal.	
Post-requisito	R2. Permitir entrar al portal. R3. Mostrar los flujos y las tareas de la carga inicial.	

Tabla 2. Descripción del requisito R1.

Requisito R2

Precondiciones	Precondiciones	Pre-requisito
	El flujo de carga inicial ha sido completado por la entidad que intenta acceder al portal.	R1. Verificar configuración de la carga inicial.
Descripción	Permite a la entidad acceder al portal en caso de se haya configurado la carga inicial correspondiente.	
Post-requisito		

Tabla 3. Descripción del requisito R2.

Requisito R3

Precondiciones	Precondiciones	Pre-requisito
	El flujo de carga inicial no ha sido completado por la entidad que intenta acceder al portal.	R1. Verificar configuración de la carga inicial.
Descripción	Carga los flujos y las tareas de la carga inicial para su ejecución pertenecientes a la entidad.	
Post-requisito	R2. Permitir entrar al portal.	

Tabla 4. Descripción del requisito R3.

Requisito R4

	Precondiciones	Pre-requisito
Precondiciones	El usuario debe actualizar el panel que contiene los flujos y las tareas a no ser el primer instante del proceso.	R3. Cargar los flujos y las tareas de la carga inicial.
Descripción	Chequea las precondiciones y postcondiciones de cada tarea.	
Post-requisito		

Tabla 5. Descripción del requisito R4.

Requisito R5

	Precondiciones	Pre-requisito
Precondiciones		R4. Verificar las precondiciones y postcondiciones de cada tarea.
Descripción	No le permite al usuario ejecutar las tareas que dependan de la ejecución de otras.	
Post-requisito		

Tabla 6. Descripción del requisito R5.

2.2.1 No funcionales

Usabilidad (USB)

1. La solución podrá ser ejecutada por cualquier persona que posea conocimientos básicos en el manejo de la computadora.
2. La navegación será intuitiva.
3. Las acciones no estarán a más de 3 clics del formulario inicial.
4. La interfaz debe visualizar en cada momento el estado del flujo.

Rendimiento (REN)

1. Los tiempos de respuesta y velocidad de procesamiento de la información estarán entre los 15 y 20 segundos, debido que el componente debe procesar una gran cantidad de servicios pertenecientes a cada uno de los subsistemas.

Seguridad (SEG)

1. La autenticación será gestionada por el subsistema de seguridad (contraseña de acceso).

Portabilidad (POR)

1. La solución es multiplataforma, haciendo énfasis en Linux y Windows.
2. Navegador Mozilla Firefox.

Soporte (SOP)

1. La solución contará, antes de su puesta en marcha, con un período de pruebas. Se le dará mantenimiento, configuración y se brindará el servicio de instalación.

Políticos – Culturales (CUL)

1. La solución solo podrá ser utilizada en territorio cubano y por las entidades autorizadas por el Ministerio de las Fuerzas Armadas Revolucionarias (MINFAR).
2. La solución no debe contener palabras en otros idiomas.
3. La solución debe respetar los términos empleados normalmente por los especialistas en el tema de la esfera que se automatiza.

Legales (LEG)

1. La solución está avalada por documentos rectores emitidos en el país para la certificación y validación de los sistemas contables:
 - La Resolución Orden #4 del Ministro de las Fuerzas Armadas Revolucionarias.

Software (SFT)

Requerimientos mínimos tecnológicos de software.

Para el cliente:

1. Sistema operativo Windows 98 o superior, o Linux.

Para el servidor:

1. Sistema operativo Windows Advancer Server (2000 o superior) o Linux en cualquiera de sus distribuciones.
2. Un servidor Apache 2.0 o superior con módulo PHP 5.0 disponible. Este debe estar configurado con la extensión “pgsql” incluida.
3. Un servidor de base de datos PostgreSQL 8.0 o superior.

Hardware (HDW)

Para el cliente:

1. Requerimientos mínimos: Procesador Pentium II a 133Mhz con 128 Mb de memoria RAM.
2. Tarjeta de red.

Para el servidor:

1. Requerimientos mínimos: Procesador Pentium III a 1GHz de velocidad de procesamiento y 1Gb de memoria RAM.
2. Al menos 40Gb de espacio libre en disco duro.
3. Tarjeta de red.

Requisitos de interfaz

1. Las tareas terminadas tienen un ícono de color verde.
2. Las tareas pendientes tienen un ícono de color amarillo.
3. Cada escenario de las tareas debe estar encabezado por el ícono y el nombre del flujo al que pertenecen.
4. Los flujos con todas las tareas terminadas tienen un ícono de color verde.
5. Los flujos con alguna tarea pendiente tienen un ícono de color amarillo.
6. Los flujos que tengan todas las tareas terminadas correspondientes a los subflujos del mismo tienen un ícono de color verde.
7. Los flujos que tengan alguna tarea pendiente correspondiente a uno de sus subflujos tienen un ícono de color amarillo.
8. Las tareas que no se puedan ejecutar en ese instante del flujo están desactivas.
9. El usuario podrá ver una breve descripción tanto del flujo como de la tarea que desee.

2.2 Prototipo de interfaz

Se utilizan los prototipos de interfaz para representar las funcionalidades del componente y validarlas con los clientes, así como la manera en que se gestionará la información.

La interfaz constará con un panel en la parte izquierda el cual contendrá los flujos. En el panel derecho se mostrará toda la información correspondiente a la tarea que se encuentre en ejecución, así como en la parte superior del mismo el nombre de la tarea en ejecución y del flujo al cual pertenece.

El color predominante es el azul, el cual permite muy buena visibilidad en las composiciones gráficas en general y en las aplicaciones web en concreto. Es utilizado ampliamente como color corporativo, por la seriedad y confianza que inspira y admite buenas gradaciones, pudiendo ser el color predominante en una página.

El diseño simétrico de la interfaz sugiere estabilidad y equilibrio, resultando estético, ordenado, atractivo y agradable de contemplar.

La simetría usada no es del todo exacta, ya que una simetría demasiado perfecta hace parecer las composiciones artificiales y premeditadas. Pequeñas variaciones en la distribución simétrica dan ese toque de ruptura que hace su contemplación más amena y natural.

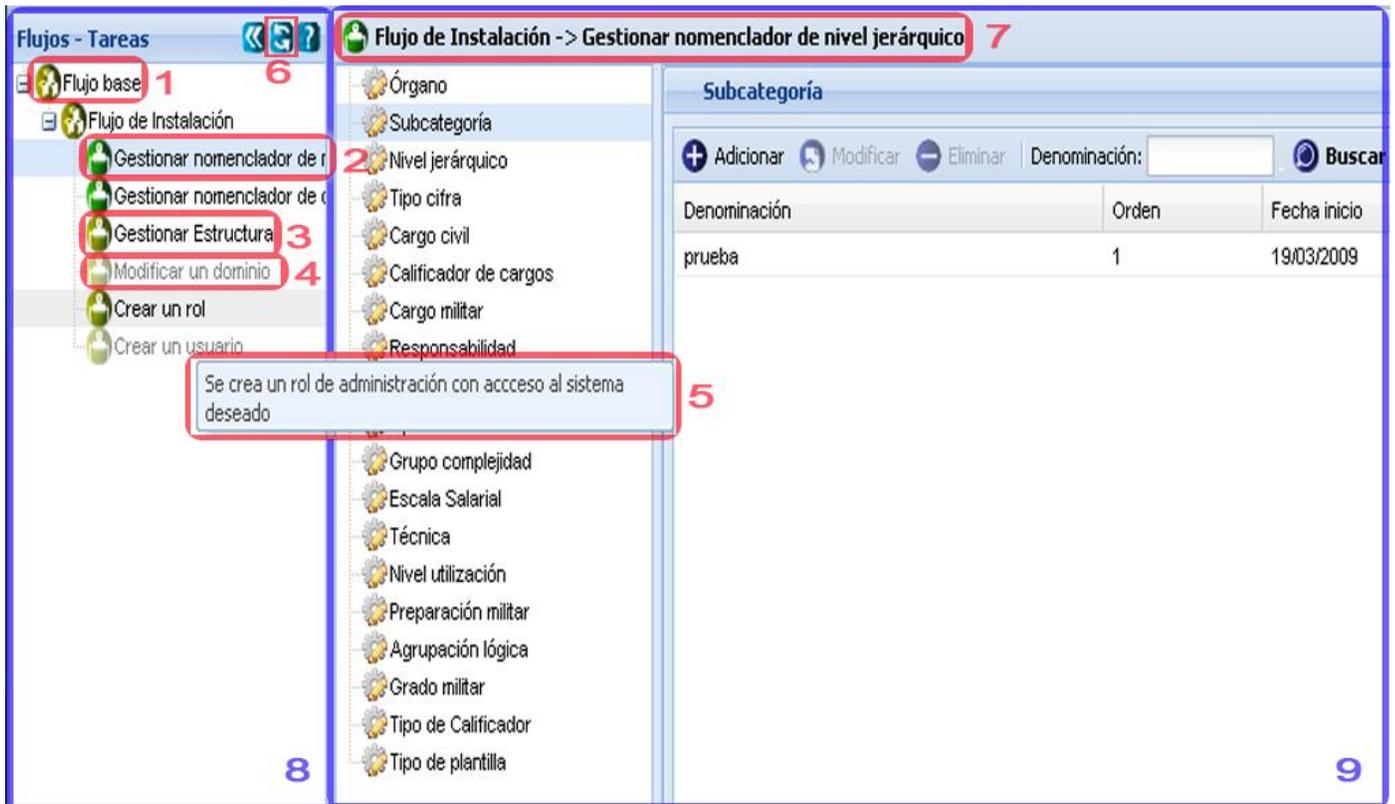


Figura 10. Prototipo de interfaz.

Para un mejor entendimiento del diseño de interfaz, en el siguiente epígrafe se describen cada una de sus partes fundamentales.

2.4.1 Descripción del prototipo de interfaz

A cada descripción le corresponde el mismo número contenido en la Figura 9.

1. Flujo con al menos una de sus tareas incompletas. El flujo tendrá el mismo ícono pero de color verde cuando todas sus tareas y sub-flujos hayan sido completados.
2. Tarea completada.
3. Tarea que no ha sido completada.
4. Tarea desactivada ya que no se han cumplido las precondiciones necesarias para que el usuario pueda ejecutarla.

5. Ventana flotante que aparece automáticamente cuando el cursor se encuentre encima de un flujo o tarea, portando la descripción del mismo.
6. Botón que actualiza el panel que contiene todos los flujos y las tareas, chequeando las precondiciones y postcondiciones de cada tarea contenida en el panel.
7. Nombre del flujo y tarea correspondiente que se encuentra en ejecución.
8. Panel que contiene los flujos y las tareas.
9. Panel que muestra toda la información correspondiente a la tarea seleccionada.

2.5 Patrones utilizados

Los patrones de flujo de trabajo que se aplicaron en la solución son descritos a continuación.

Terminación Implícita (Implicit Termination)

Describe que un subproceso debe terminar cuando no haya nada más que hacer. En otras palabras, que no haya actividades activas en el flujo de trabajo. (Aalst)

Este patrón se aplica, ya que no es posible acceder al portal mientras no se haya completado el flujo de carga inicial. Todas sus tareas deben haber sido configuradas.

División Paralela (Parallel Split)

Describe un punto en el proceso de flujo de trabajo donde un único hilo de control se divide en múltiples hilos de control, los cuales pueden ser ejecutados en paralelo, permitiendo a las actividades ser ejecutadas simultáneamente o en cualquier orden. (Aalst)

La aplicación de este patrón en la solución se evidencia cuando la culminación de una tarea es la única precondición de otras. Una vez completada la ejecución de la misma, las tareas dependientes se activarán permitiendo al usuario ejecutar la que desee.

Unión Sincronizada (Synchronizing Merge)

Describe un punto en el proceso de flujo de trabajo donde múltiples hilos convergen en un único hilo de control. (Aalst)

Este patrón se aplica cuando una tarea tiene por precondición la culminación de varias tareas. Una vez completada la ejecución de las mismas, la tarea dependiente se activará permitiendo al usuario acceder a ella.

2.6 Diagrama de componentes

En el diagrama de componentes (ver Figura 11), dentro del paquete que recibe por nombre “*Otros subsistemas*” están incluidos el resto de los subsistemas de los cuales se consumen servicios, no fueron representados ya que son un gran número de ellos y la dimensión del diagrama sería mucho mayor. La relación que existe entre los componentes de estos subsistemas y la carga inicial es igual a la del resto.

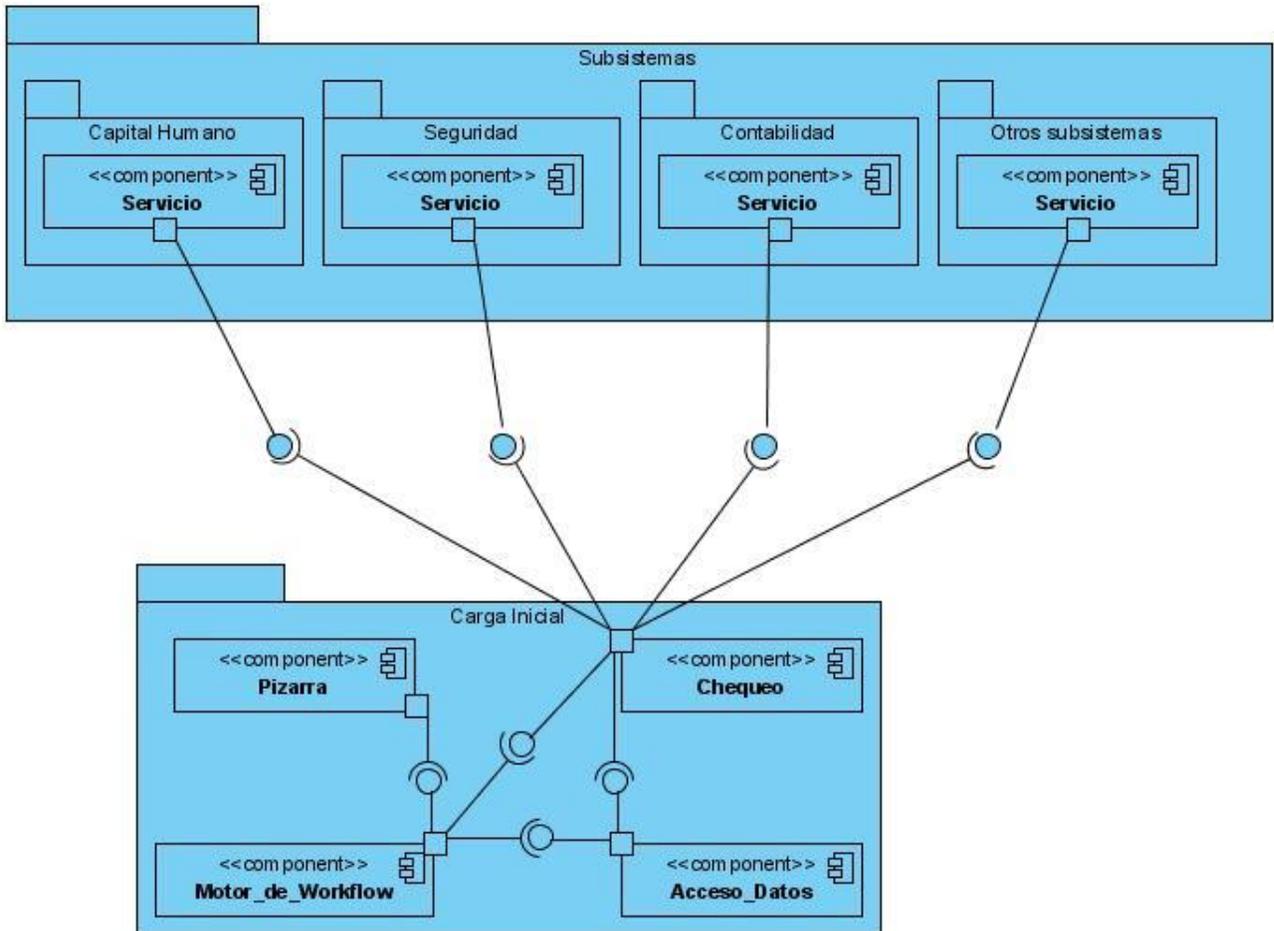


Figura 11. Diagrama de componentes

En el Anexo 6: Direcciones de ejecución y chequeo de las tareas. se muestra una tabla que contiene las direcciones de los servicios de ejecución y chequeo de las tareas.

2.7 Diseño de clases

Los diagramas de clases de diseño junto a los demás artefactos que se generan en este flujo de trabajo son el centro de atención al final de la fase de elaboración y el comienzo de las iteraciones de construcción. Esto contribuye a una arquitectura estable y sólida y crea un plano del modelo de implementación.

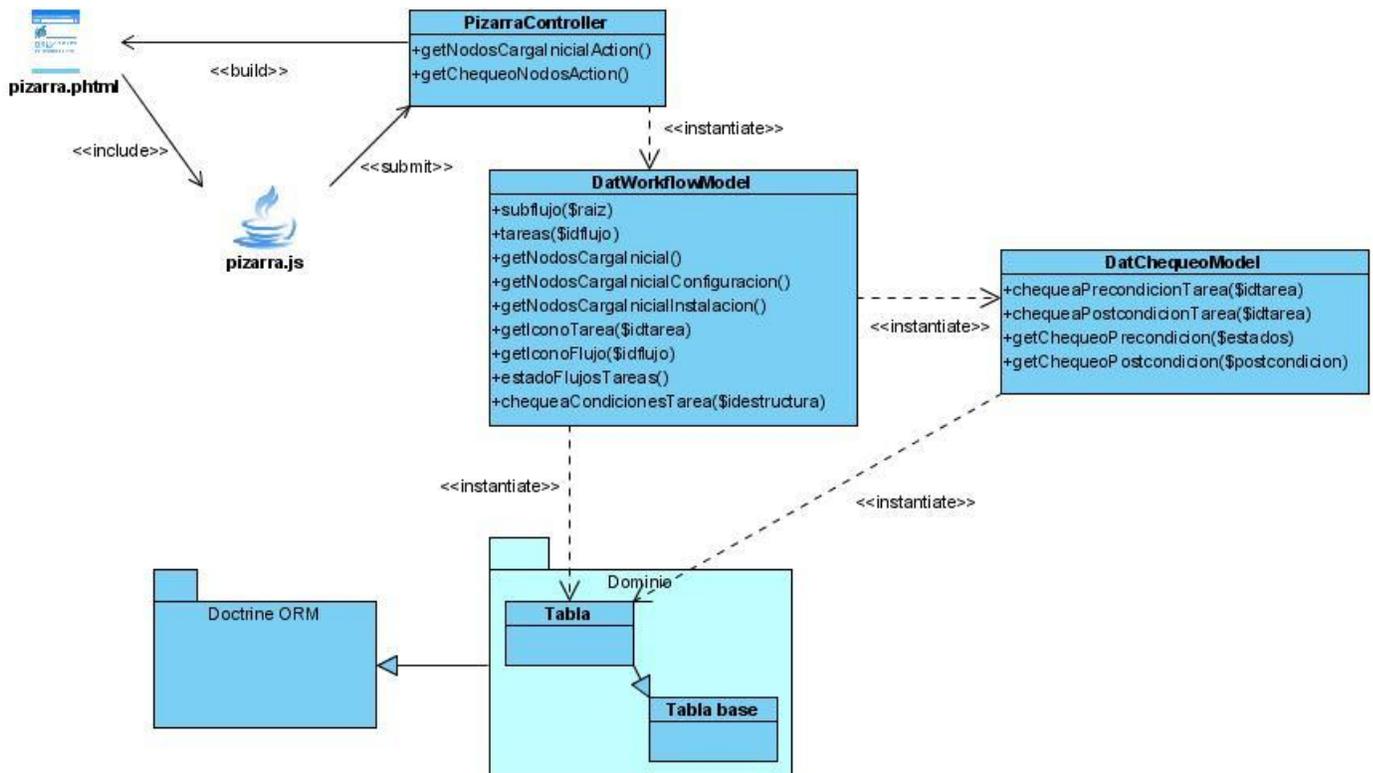


Figura 12. Diagrama de clases.

Las clases de dominio contenidas en el paquete *Dominio* (ver Figura 12) se reflejan en el siguiente diagrama de clases.

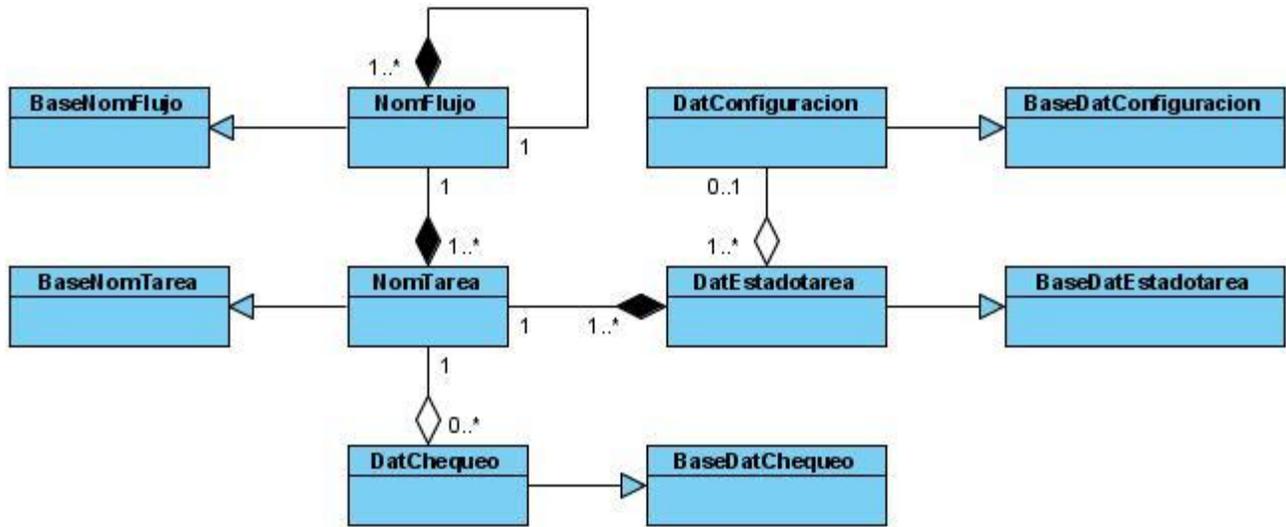


Figura 13. Diagrama de clases del paquete de Dominio.

Clase controladora:

Nombre	PizarraController
Tipo de clase	Controladora.
Para cada responsabilidad	
Nombre	Descripción
getNodosCargaInicialAction()	Proporciona la estructura del árbol correspondiente a un flujo.
getChequeoNodosAction()	Actualiza el árbol que contiene los flujos, para determinar el estado de cada una de las tareas en ese instante del proceso.

Tabla 7. Descripción de la clase controladora PizarraController.

Clases de negocio:

Nombre	DatWorkflowModel
Tipo de clase	Negocio.
Para cada responsabilidad	
Nombre	Descripción
subflujo(\$raiz)	Compone la estructura de todos los subflujos recursivamente.
tareas(\$idflujo)	Proporciona las tareas pertenecientes a un flujo determinado.

getNodosCargaInicial()	En dependencia de la estructura selecciona el flujo a cargar.
getNodosCargaInicialConfiguracion()	Proporciona la estructura del árbol perteneciente al flujo de configuración.
getNodosCargaInicialInstalacion()	Proporciona la estructura del árbol perteneciente al flujo de instalación.
getIconoTarea()	Asigna un ícono a cada tarea de acuerdo a su estado.
getIconoFlujo()	Chequea todas las tareas y subflujos correspondientes a cada flujo y permite asignarle un ícono dependiente del estado de las tareas pertenecientes a él.
estadoFlujosTareas()	Actualiza cada tarea del árbol que se encuentren en el flujo.
chequeaCondicionesTarea(\$idestructura)	Chequea las precondiciones y postcondiciones de cada tarea.

Tabla 8. Descripción de la clase del negocio DatWorkflowModel.

Nombre	DatChequeoModel
Tipo de clase	Negocio.
Para cada responsabilidad	
Nombre	Descripción
chequeaPrecondicionTarea(\$idtarea)	Chequea las precondiciones establecidas en la base de datos para cada tarea.
chequeaPostcondicionTarea(\$idtarea)	Chequea las postcondiciones establecidas en la base de datos para cada tarea.
getChequeoPrecondicion(\$estados)	Chequea las precondiciones establecidas en la base de datos para cada tarea.

Tabla 9. Descripción de la clase del negocio DatChequeoModel.

Clases entidad:

Las clases entidad se encuentran incluidas en el paquete *Domino* del diagrama de clases (ver Figura 12)

Nombre	NomTarea
Tipo de clase	Entidad.
Descripción	
Permite el acceso a la tabla de la base de datos dat_tarea, que contiene la información correspondiente a las tareas.	

Tabla 10. Descripción de la clase entidad DatTarea.

Nombre	NomFlujo
Tipo de clase	Entidad.
Descripción	
Permite el acceso a la tabla de la base de datos dat_flujo, que contiene la información correspondiente a los flujos.	

Tabla 11. Descripción de la clase entidad DatFlujo.

Nombre	DatEstadotarea
Tipo de clase	Entidad.
Descripción	
Permite el acceso a la tabla de la base de datos dat_estadotarea, que contiene la información correspondiente al estado de las tareas.	

Tabla 12. Descripción de la clase entidad DatEstadotarea.

Nombre	DatChequeo
Tipo de clase	Entidad
Descripción	
Permite el acceso a la tabla de la base de datos dat_chequeo, que contiene la información correspondiente a las precondiciones y postcondiciones de cada tarea.	

Tabla 13. Descripción de la clase entidad DatChequeo.

Nombre	DatConfiguracion
Tipo de clase	Entidad
Descripción	
Permite el acceso a la tabla de la base de datos dat_configuracion, que contiene la información correspondiente a las precondiciones y postcondiciones de cada tarea.	

Tabla 14. Descripción de la clase entidad DatConfiguracion.

2.8 Modelo de datos

Los modelos de datos (MD), son el mecanismo formal para representar los datos de manera general y sistemática. Establecen una relación entre el mundo real y la información almacenada físicamente en la base de datos (BD). (García Mato, 1999)

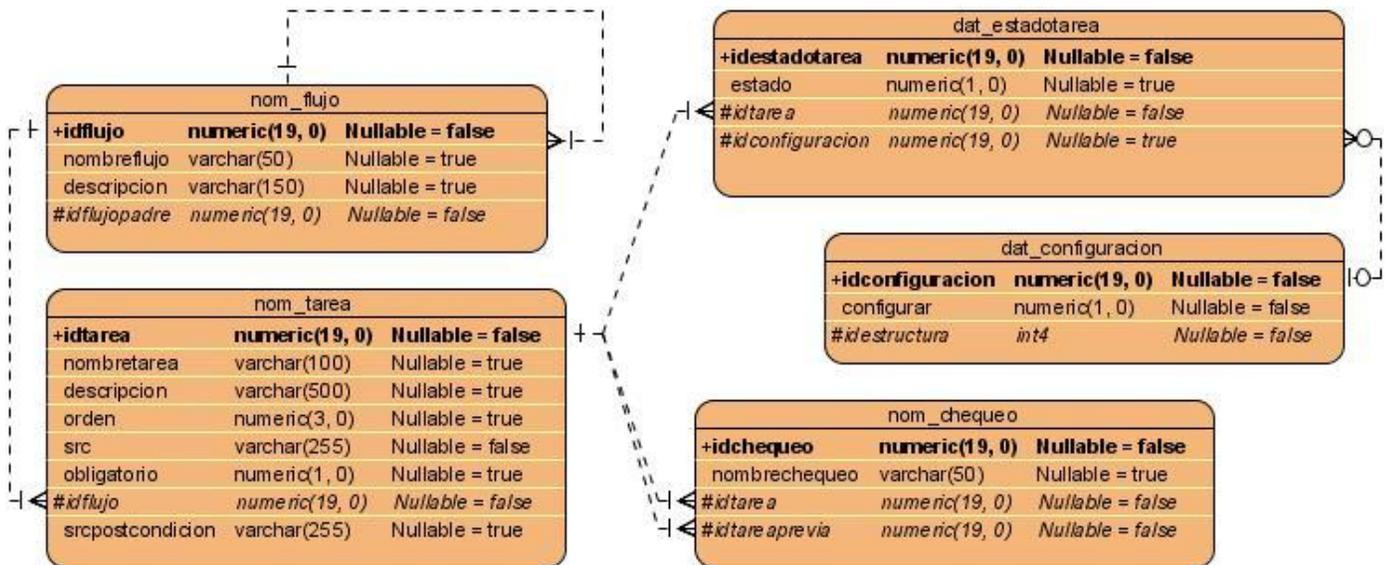


Figura 14. Modelo de datos

Es necesario hacer una breve descripción de cada una de las tablas para un mejor entendimiento del MD.

Tabla *dat_estadotarea*

Almacena el estado de cada una de las tareas. Se consulta esta tabla cada vez que se desee conocer el estado de cualquier tarea en cualquier instante del flujo, la misma debe actualizarse antes de poder pasar a un siguiente estado.

Tabla *dat_configuracion*

La información referente a la configuración de las tareas se almacena en esta tabla para optimizar el tiempo de búsqueda del algoritmo que chequea las postcondiciones. En ella se almacena el estado de configuración de cada tarea. Además es consultada para obtener la configuración de las tareas pertenecientes a una estructura.

Tabla *nom_flujo*

Almacena la información correspondiente al flujo, incluyendo una descripción del mismo.

Tabla *nom_tarea*

Almacena la información de las tareas, incluyendo la dirección de su origen y de su postcondición. Además del identificador del flujo al cual pertenece la tarea, una descripción también es almacenada en la tabla.

Tabla *nom_chequeo*

La información correspondiente a las precondiciones se almacena en esta tabla. Todas las tareas que deben estar completadas para que se permita la ejecución de otra, se definen en ella.

2.9 Casos de prueba

Los casos de pruebas son un conjunto de entradas con datos de prueba, unas condiciones de ejecución, y unos resultados esperados cuyo propósito es Identificar y comunicar las condiciones que se llevarán a cabo en la prueba. Los casos de la prueba son necesarios para verificar la aplicación exitosa y aceptable de los requisitos del producto.

La descripción de los casos de prueba del componente se realizó por escenarios los cuales son detallados en cada uno de los diseños de casos de prueba (DCP) siguientes.

DCP Verificar configuración de la carga inicial.

1. Condiciones de ejecución.

- La BD del sistema debe contener solamente los datos generados una vez instalado el mismo.
- Debe existir una estructura que haya completado el flujo de carga inicial.
- Debe existir una estructura que no haya completado el flujo de carga inicial.

1.1 Requisitos a probar.

Nombre del requisito	Descripción General	Escenarios de pruebas	Flujo del escenario
R1: Verificar configuración de la carga inicial.	Verifica la configuración de la carga inicial para la entidad que intenta acceder al portal.	EP 1.1: Acceder al portal mediante una estructura que no haya completado el flujo de carga inicial.	- El usuario elige una estructura que haya configurado la carga inicial.
		EP 1.2: Acceder al portal mediante una estructura que haya completado el flujo de carga inicial.	- El usuario elige una estructura que no haya configurado la carga inicial.

Tabla 15. DCP verificar configuración de la carga inicial.

1.1.1 Descripción de variable.

Nombre del campo	Clasificación	Puede ser nulo	Descripción
NA.	NA.	NA.	NA.

1.1.2 Juego de datos a probar.

Id del escenario	Escenario	Nombre	Respuesta del sistema	Resultado de la prueba
EP 1.1	Acceder al portal mediante una estructura que no haya completado el flujo de carga inicial.	NA.	Se abre el portal.	Se abre el portal.
EP 1.2	Acceder al portal mediante una estructura que no haya completado el flujo de carga inicial.	NA.	Se abre el flujo de carga inicial.	Se abre el flujo de carga inicial.

DCP permitir acceder al portal.

1. Condiciones de ejecución.

- La BD del sistema debe contener solamente los datos generados una vez instalado el mismo.
- Debe existir una estructura que haya completado el flujo de carga inicial.

1.1 Requisitos a probar.

Nombre del requisito	Descripción General	Escenarios de pruebas	Flujo del escenario
R2: Permitir acceder al portal	Permite al usuario acceder al portal.	EP 1.1: Acceder al portal mediante una estructura que haya completado el flujo de carga inicial.	- El usuario elige una estructura que haya configurado la carga inicial.

Tabla 16. DCP permitir acceder al portal.

1.1.1 Descripción de variable.

Nombre del campo	Clasificación	Puede ser nulo	Descripción
NA.	NA.	NA.	NA.

1.1.2 Juego de datos a probar.

Id del escenario	Escenario	Nombre	Respuesta del sistema	Resultado de la prueba
EP 1.1	Acceder al portal mediante una estructura que haya completado el flujo de carga inicial.	NA.	Se abre el portal.	Se abre el portal.

DCP cargar los flujos y las tareas de la carga inicial.

1. Condiciones de ejecución.

- La BD del sistema debe contener solamente los datos generados una vez instalado el mismo.
- Debe existir una estructura que no haya completado el flujo de carga inicial.

1.1 Requisitos a probar.

Nombre del requisito	Descripción General	Escenarios de pruebas	Flujo del escenario
R3: Cargar los flujos y tareas de la carga inicial.	Carga los flujos y tareas de la carga inicial para su ejecución.	EP 1.1: Acceder al portal mediante una estructura que no haya completado el flujo de carga inicial.	- El usuario elige una estructura que no haya configurado la carga inicial.

Tabla 17. DCP cargar los flujos y las tareas de la carga inicial.

1.1.1 Descripción de variable.

Nombre del campo	Clasificación	Puede ser nulo	Descripción
NA.	NA.	NA.	NA.

1.1.2 Juego de datos a probar.

Id del escenario	Escenario	Nombre	Respuesta del sistema	Resultado de la prueba
EP 1.1	Acceder al flujo de carga inicial.	NA.	Se abre el flujo de carga inicial.	Se abre el flujo de carga inicial.

DCP verificar las precondiciones y postcondiciones de cada tarea.

1. Condiciones de ejecución.

- La BD del sistema debe contener solamente los datos generados una vez instalado el mismo.
- Debe existir una estructura que no haya completado el flujo de carga inicial.

1.1 Requisitos a probar.

Nombre del requisito	Descripción General	Escenarios de pruebas	Flujo del escenario
R4: Verificar las precondiciones y postcondiciones de cada tarea.	Verifica las precondiciones y postcondiciones de cada tarea.	EP 1.1: Ejecutar una tarea que su ícono sea de color amarillo.	<ul style="list-style-type: none"> - El usuario elige una tarea que no haya sido completada y que la ejecución de la misma sea la única precondición de alguna tarea desactivada. - Se ejecuta la tarea. - Se presiona el botón Actualizar.

Tabla 18. DCP verificar las precondiciones y postcondiciones de cada tarea.

1.1.1 Descripción de variable.

Nombre del campo	Clasificación	Puede ser nulo	Descripción
NA.	NA.	NA.	NA.

1.1.2 Juego de datos a probar.

Id del escenario	Escenario	Nombre	Respuesta del sistema	Resultado de la prueba
EP 1.1	Acceder al flujo de carga inicial.	NA.	<ul style="list-style-type: none"> - La tarea ejecutada tiene un ícono de color verde. - La tarea que estaba desactivada está 	<ul style="list-style-type: none"> - La tarea ejecutada tiene un ícono de color verde. - La tarea que estaba desactivada está

			activada y su ícono es de color amarillo.	activada y su ícono es de color amarillo.
--	--	--	---	---

DCP permitir ejecutar solamente las tareas que tengan cumplidas sus precondiciones.

1. Condiciones de ejecución.

- La BD del sistema debe contener solamente los datos generados una vez instalado el mismo.
- Debe existir una estructura que no haya completado el flujo de carga inicial.

1.1 Requisitos a probar.

Nombre del requisito	Descripción General	Escenarios de pruebas	Flujo del escenario
R5: Permitir ejecutar solamente las tareas que tengan cumplidas sus precondiciones.	Estarán activas solamente las tareas que tengan cumplidas todas sus precondiciones, en caso contrario estarán desactivadas y el usuario no podrá ejecutarlas.	EP 1.1: Acceder al portal mediante una estructura que no haya completado el flujo de carga inicial.	- El usuario elige una estructura que no haya configurado la carga inicial. - Se presiona sobre una tarea que se encuentre desactivada.
		EP 1.2: Acceder al portal mediante una estructura que no haya completado el flujo de carga inicial.	- El usuario elige una estructura que no haya configurado la carga inicial. - Se presiona sobre una tarea que se encuentre activada y que su ícono sea de color amarillo.

Tabla 19. DCP permitir ejecutar solamente las tareas que tengan cumplidas sus precondiciones.

1.1.1 Descripción de variable.

Nombre del campo	Clasificación	Puede ser nulo	Descripción
NA.	NA.	NA.	NA.

1.1.2 Juego de datos a probar.

Id del escenario	Escenario	Nombre	Respuesta del sistema	Resultado de la prueba
EP 1.1	Acceder al flujo de carga inicial.	NA.	- No ocurre nada.	- No ocurre nada.
EP 1.2	Acceder al flujo de carga inicial.	NA.	- El usuario puede ejecutar la tarea.	- El usuario puede ejecutar la tarea.

Se ejecutaron 7 iteraciones de pruebas y una funcional, en las 3 primeras el promedio de errores implícitos por codificación estaba dado en el orden de las 6 unidades, el promedio de errores implícito por el consumo de servicios estaba dado en el orden de las 33 unidades, en las 3 iteraciones restantes el promedio de errores implícitos por codificación estaba dado en el orden de las 1,5 unidades, el promedio de errores implícito por el consumo de servicios estaba dado en el orden de las 12,75 unidades. En la iteración funcional se emitieron 7 solicitudes de cambio y 2 no conformidades, las cuales fueron resueltas y aprobadas en una octava iteración asociada a la prueba funcional.

Conclusiones

Entre los fundamentales aspectos incluidos en el proceso de análisis, se desarrolló el levantamiento de los requisitos funcionales y no funcionales de la solución, necesarios para lograr un óptimo desempeño de la misma. Se diseñó el prototipo de interfaz relacionado con los requisitos y se generaron distintos artefactos de implementación, como el diagrama de componentes y modelo de datos. Por último, para la comprobación de las funcionalidades implementadas, se elaboraron los casos de prueba los cuales reflejan un comportamiento óptimo de la solución, obteniéndose como resultado de las pruebas una correcta respuesta del sistema, cumpliendo así los requisitos establecidos.

EVALUACIÓN DE LA SOLUCIÓN

Introducción

En este capítulo se hace referencia a métricas que son aplicadas para validar el diseño de software en cuanto a su calidad. Se definen las métricas que se aplicaron a la solución y se realiza una valoración de los resultados. Por último se aborda la matriz de inferencia de indicadores de calidad.

3.1 Métricas que evalúan el diseño

Cuando se planifica un proyecto se tiene que obtener estimaciones del costo y esfuerzo humano requerido por medio de las mediciones de software que se utilizan para recolectar los datos cualitativos acerca del software y sus procesos para aumentar su calidad. (Universidad de Guadalajara, 2008)

En la mayoría de los desafíos técnicos, las métricas ayudan a entender tanto el proceso técnico que se utiliza para desarrollar un producto, como el propio producto. (Universidad de Guadalajara, 2008)

3.1.1 Métricas de software

Aunque los términos medida, medición y métricas se utilizan a menudo indistintamente, existe gran confusión a la hora de referirnos a ellos. Dentro del contexto de la ingeniería del software, una medida *proporciona una indicación cuantitativa de extensión, cantidad, dimensiones, capacidad y tamaño de algunos atributos de un proceso o producto.* (Pressman, 2002)

Las métricas de software se centran en cuantificar tanto la complejidad, como la funcionalidad y eficiencia inmersa en el desarrollo de software, inclinando sus objetivos a mejorar la comprensión de la calidad del producto, estimar efectividad del proceso y mejorar la calidad del trabajo.

3.1.2 Métricas orientadas a objetos

Las métricas orientadas a objetos se centran en métricas que se pueden aplicar a las características de encapsulamiento, ocultamiento de información, herencia y técnicas de abstracción de objetos que hagan única a esa clase.

Chidamber y Kemerer proponen una familia de medidas para desarrollos orientados a objetos (Chidamber, y otros):

- Métodos ponderados por clase (MPC): Tamaño y complejidad
- Profundidad árbol de herencia (PAH): Tamaño
- Número de descendientes (NDD): Tamaño, acoplamiento y cohesión
- Acoplamiento entre clases (ACO): Acoplamiento
- Respuesta para una clase (RPC): Comunicación y complejidad
- Carencia de cohesión en los métodos (CCM): Cohesión interna

Estas métricas, en líneas generales, permiten averiguar cuán bien están definidas las clases y el sistema, lo cual tiene un impacto directo en el mantenimiento del mismo, tanto por la comprensión de lo desarrollado como por la dificultad de modificarlo con éxito.

3.1.3 Métricas orientadas a clases

El Software Orientado a Objetos (OO) es fundamentalmente distinto del software que se desarrolla utilizando métodos convencionales. Las métricas para sistemas OO deben de ajustarse a las características que distinguen el software OO del software convencional. Estas métricas hacen hincapié en el encapsulamiento, la herencia, complejidad de clases y polimorfismo. Por lo tanto las métricas OO se centran en métricas que se pueden aplicar a las características de encapsulamiento, ocultamiento de información, herencia y técnicas de abstracción de objetos que hagan única a esa clase. Como en todas las métricas los objetivos principales de las métricas OO se derivan del software convencional:

Comprender mejor la calidad del producto, estimar la efectividad del proceso y mejorar la calidad del trabajo realizado a nivel del proyecto.

3.1.4 Métricas aplicadas a la solución propuesta

Las métricas escogidas para la solución propuesta fueron de la serie Chidamber y Kemner (Chidamber, y otros):

- Profundidad de Herencia (PH): Está dado por la profundidad en la herencia de las clases heredadas de un nodo padre.
- Número de Descendiente (ND): Está dado por la cantidad de clases que heredan de un padre.
- Respuesta para una Clase (RC): Está dado por el número de relaciones de uso de una clase con otras.

Se utilizaron de la serie definida por Lorenz y Kidd (Lorenz, y otros, 1994) dos métricas:

- Tamaño Operacional de Clases (TOC): Esta dado por el número de métodos asignados a una clase.
- Número de Operaciones Redefinidas para una Sub-Clase (NOR): Está dado por la cantidad de operaciones redefinidas en cada clase hija.

Los atributos de calidad que abarcan dichas métricas son:

Responsabilidad: Consiste en la responsabilidad asignada a una clase en un marco de modelado de un dominio o concepto, de la problemática propuesta.

Complejidad del diseño: Consiste en la complejidad que posee una estructura de diseño de clases.

Complejidad de implementación: Consiste en el grado de dificultad que tiene implementar un diseño de clases determinado.

Reutilización: Consiste en el grado de reutilización que esté presente en una clase o estructura de clase, dentro de un diseño de software.

Acoplamiento: Consiste en el grado de dependencia o interconexión de una clase o estructura de clase, con otras, está muy ligada a la característica de Reutilización.

Complejidad del mantenimiento: Consiste en el grado de esfuerzo necesario a realizar para desarrollar un arreglo, una mejora o una rectificación de algún error de un diseño de software. Puede influir indirecta, pero fuertemente en los costes y la planificación del proyecto.

Cantidad de pruebas: Consiste en el número o el grado de esfuerzo para realizar las pruebas de calidad (unidad) del producto (componente, módulo, clase, conjunto de clases, etc.) diseñado.

Nivel de cohesión: Consiste en el grado de especialización de las clases concebidas para modelar un dominio o concepto específico.

Abstracción del diseño: Consiste en la capacidad de modelar lo más cercano posible a la realidad un concepto o dominio determinado.

Para evaluar la calidad del diseño y la relación con los atributos de calidad del componente Carga Inicial se utilizaron las siguientes métricas:

Tamaño Operacional de Clases (TOC)

Atributo.	Modo de afectación.
Responsabilidad.	Un aumento del TOC implica un aumento de la responsabilidad asignada a la clase.
Complejidad de implementación.	Un aumento del TOC implica un aumento de la complejidad de implementación de la clase.
Reutilización.	Un aumento del TOC implica una disminución en el grado de reutilización de la clase.

Tabla 20. Tamaño de clases (TOC).

Respuesta para una Clase (RC)

Atributo.	Modo de afectación.
Acoplamiento.	Un aumento del RC implica un aumento del acoplamiento de la clase.
Complejidad de mantenimiento.	Un aumento del RC implica un aumento de la complejidad del mantenimiento de la clase.
Reutilización.	Un aumento del RC implica una disminución en el grado de reutilización de la clase.
Cantidad de pruebas.	Un aumento del RC implica un aumento de la cantidad de pruebas de unidad necesarias para probar una clase.

Tabla 21. Respuesta para una clase (RC).

Profundidad de Herencia (PH)

Atributo.	Modo de afectación.
Complejidad de mantenimiento.	Un aumento del PH implica una disminución de la complejidad del mantenimiento de la clase.
Complejidad de diseño.	Un aumento del PH implica una disminución de la complejidad del diseño de la clase.
Reutilización.	Un aumento del PH implica un aumento en el grado de reutilización de la clase.

Tabla 22. Árbol de profundidad de herencia (PH).

Número de Descendientes (ND)

Atributo.	Modo de afectación.
Abstracción del diseño.	Un aumento del ND puede diluir la abstracción del diseño de clases.
Cantidad de pruebas.	Un aumento del ND implica un aumento de la cantidad de pruebas de unidad necesarias para probar una clase.
Reutilización.	Un aumento del ND implica un aumento en el grado de reutilización de la clase.
Cantidad de pruebas.	Un aumento del ND implica una disminución en el nivel de cohesión de la clase.

Tabla 23. Número de descendientes (ND).

Número de Operaciones Redefinidas para una sub-clase (NOR)

Atributo.	Modo de afectación.
Abstracción del diseño.	Un aumento del NOR implica que se ha logrado una buena definición de la abstracción del diseño de clases.
Cantidad de pruebas.	Un aumento del NOR implica un aumento de la cantidad de pruebas de unidad necesarias para probar una clase.
Complejidad del mantenimiento.	Un aumento del NOR implica un aumento de la complejidad del mantenimiento de la clase.

Tabla 24. Número de operaciones redefinidas para una sub-clase (NOR).

Los resultados obtenidos aplicando los instrumentos de evaluación para las métricas anteriormente descritas y una valoración de los mismos se muestran a continuación. Estas pruebas fueron realizadas a la capa de negocio de la solución propuesta.

Resultados del instrumento de evaluación de la métrica Tamaño de Clases (TOC).

Ver instrumentos y tabla de resultados en el Anexo 1: Instrumento de medición de la métrica de Tamaño operacional de Clases (TOC).

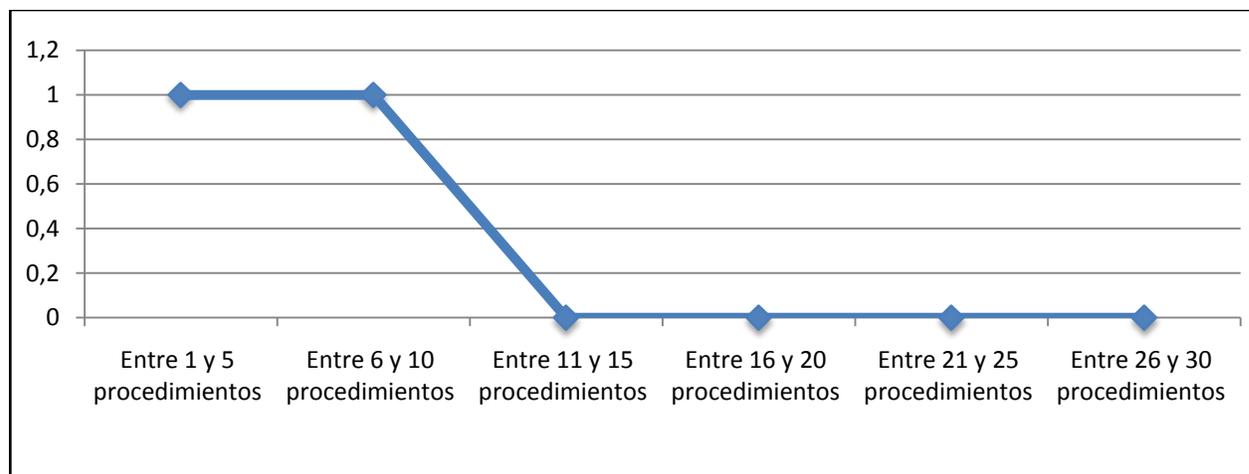


Figura 15. Representación de los resultados obtenidos en el instrumento agrupados en los intervalos definidos.

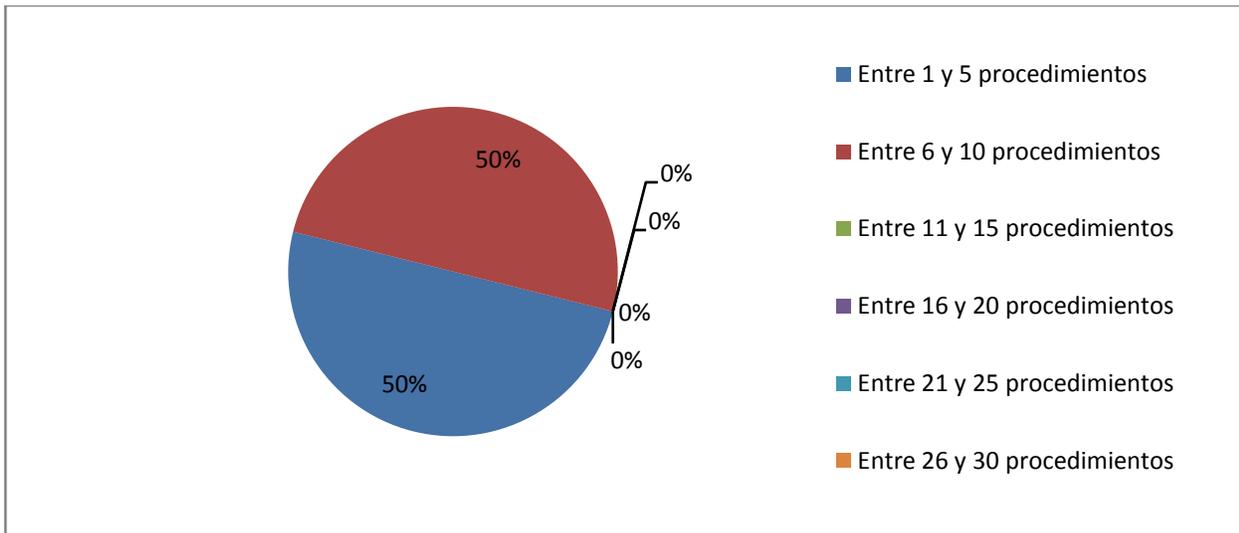


Figura 16. Representación en % de los resultados obtenidos en el instrumento agrupados en los intervalos definidos.

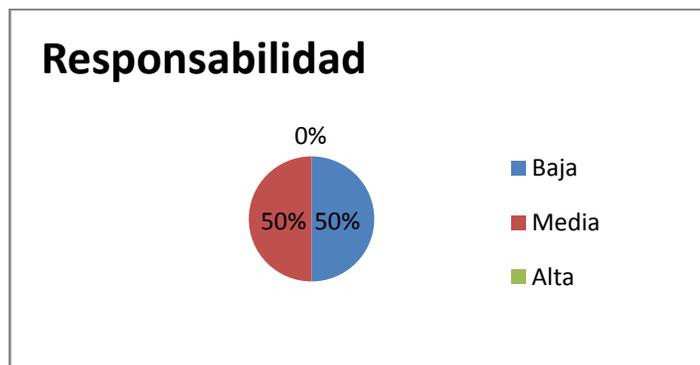


Figura 17. Representación de la incidencia de los resultados de la evaluación de la métrica TOC en el atributo Responsabilidad.



Figura 18. Representación de la incidencia de los resultados de la evaluación de la métrica TOC en el atributo Complejidad de Implementación.

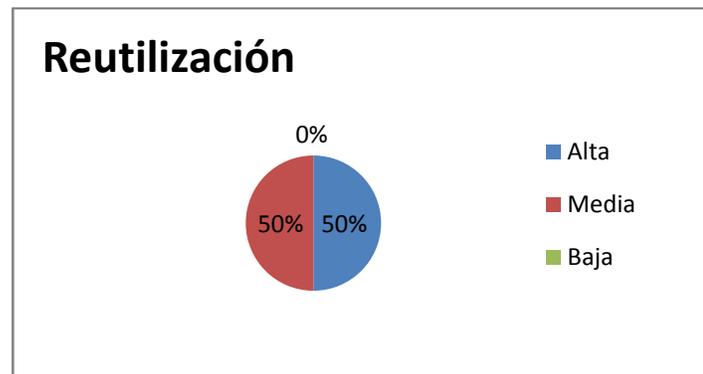


Figura 19. Representación de la incidencia de los resultados de la evaluación de la métrica TOC en el atributo Reutilización.

Haciendo un análisis de los resultados obtenidos en la evaluación del instrumento de medición de la métrica TC, se puede concluir que el diseño del componente Carga Inicial tiene una calidad favorable debido a que el 100% de las clases incluidas en este componente poseen menos cantidad de operaciones que la mitad del valor máximo registrado en las mediciones. Además, el 100% de las clases poseen evaluaciones positivas en los atributos de calidad (Responsabilidad, Complejidad de Implementación y Reutilización).

Resultado del instrumento de evaluación de la métrica Respuesta para una Clase (RC).

Ver instrumentos y tabla de resultados en el Anexo 2: Instrumento de medición de la métrica de Relaciones entre Clases (RC).

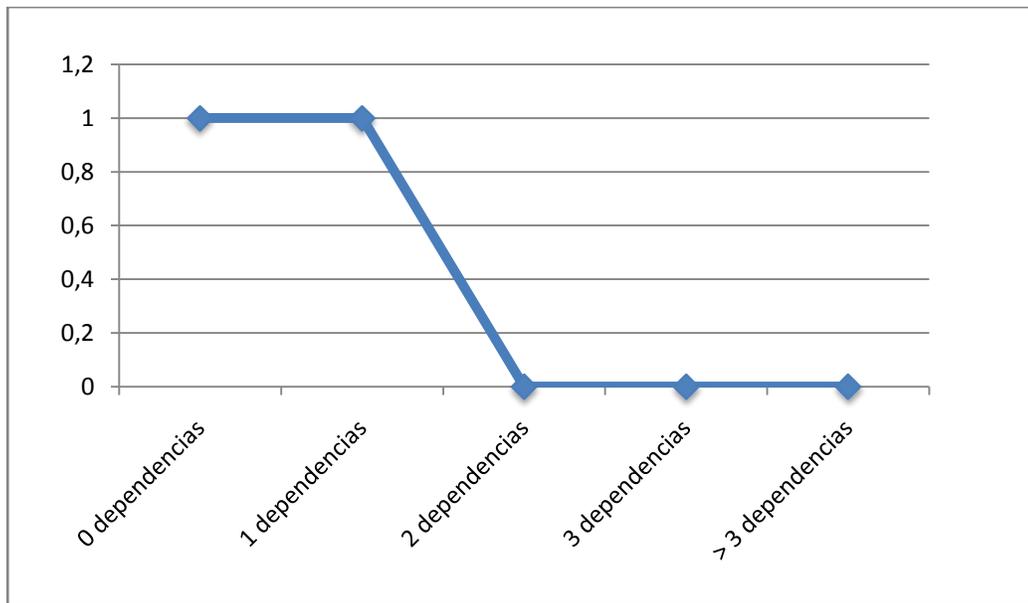


Figura 20. Representación de los resultados de la evaluación de la métrica RC agrupados por la tendencia de los valores.

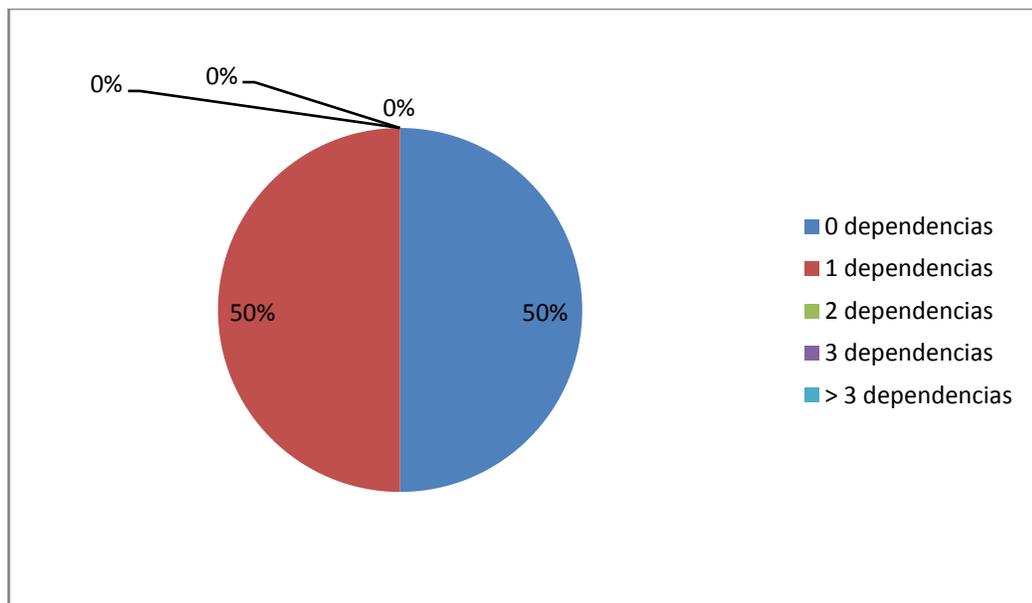


Figura 21. Representación en % de los resultados obtenidos en el instrumento agrupados en los intervalos definidos.

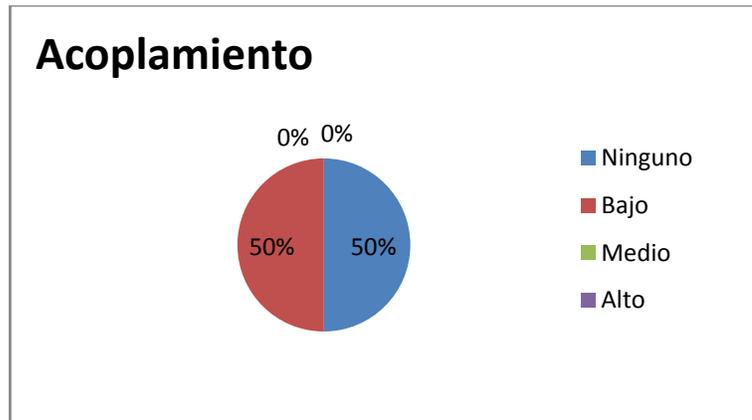


Figura 22. Representación de la incidencia de los resultados de la evaluación de la métrica RC en el atributo Acoplamiento.

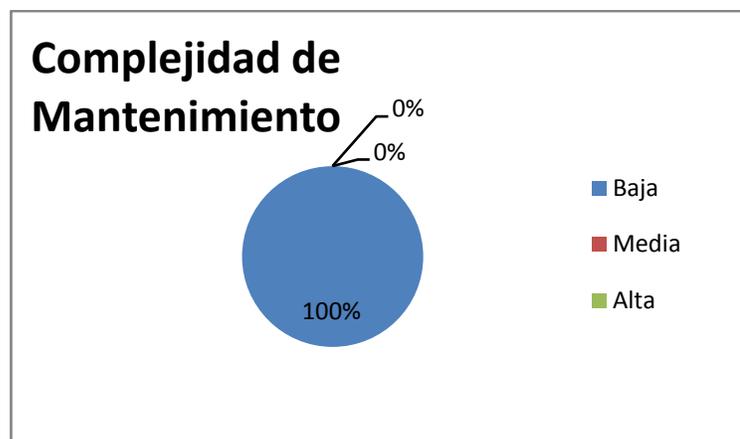


Figura 23. Representación de la incidencia de los resultados de la evaluación de la métrica RC en el atributo Complejidad de Mantenimiento.

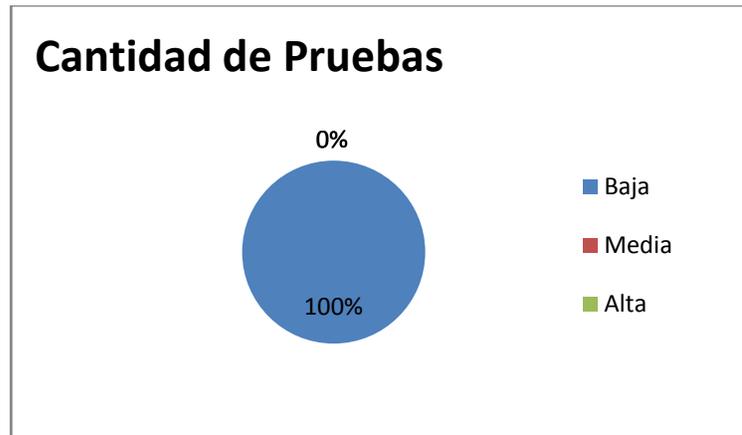


Figura 24. Representación de la incidencia de los resultados de la evaluación de la métrica RC en el atributo Cantidad de Pruebas.



Figura 25. Representación de la incidencia de los resultados de la evaluación de la métrica RC en el atributo Reutilización.

Haciendo un análisis de los resultados obtenidos en la evaluación del instrumento de medición de la métrica RC, se puede concluir que el diseño del componente Carga Inicial tiene una calidad favorable debido a que el 100% de las clases incluidas en este componente poseen menos de 2 dependencias de otras clases. Además, el 50% de las clases poseen acoplamiento y el otro 50 % un bajo acoplamiento con otras. Así mismo, los atributos de calidad Complejidad de Mantenimiento, Cantidad de Pruebas y Reutilización se comportan satisfactoriamente en el 100% de las clases.

Resultados del instrumento de evaluación de la métrica Profundidad de Herencia (PH).

Ver instrumentos y tabla de resultados en el Anexo 3: Instrumento de medición de la métrica de Profundidad de Herencia (PH).

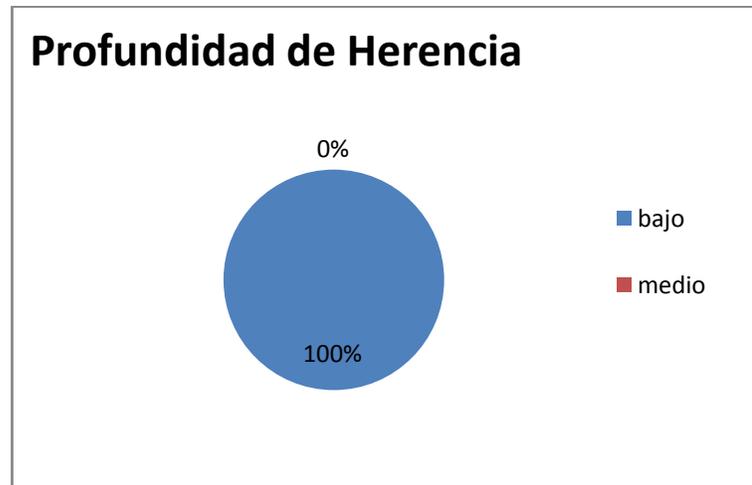


Figura 26. Representación en % de los resultados obtenidos en el instrumento agrupados por nivel.

Haciendo un análisis de los resultados obtenidos en la evaluación del instrumento de medición de la métrica PH, se puede concluir que el diseño del componente Carga Inicial tiene una calidad aceptable teniendo en cuenta que la profundidad de herencia presente en los subsistemas desarrollados es cero, el cual es el valor mínimo posible. Analizando los atributos de calidad Complejidad de Mantenimiento, Complejidad de Diseño se puede decir que, sin dudas, se tienen buenos índices, debido a que solo se cuenta como máximo con un nivel cero de profundidad en la herencia. Sin embargo el atributo Reutilización no posee indicadores positivos debido al bajo nivel de profundidad de herencia.

Resultados del instrumento de evaluación de la métrica Número de Descendientes (ND).

Ver instrumentos y tabla de resultados en el Anexo 4: Instrumento de medición de la métrica de Número de Descendientes (ND).



Figura 27. Representación de la incidencia de los resultados de la evaluación de la métrica ND en el atributo Reutilización.



Figura 28. Representación de la incidencia de los resultados de la evaluación de la métrica ND en el atributo Abstracción de la clase base.

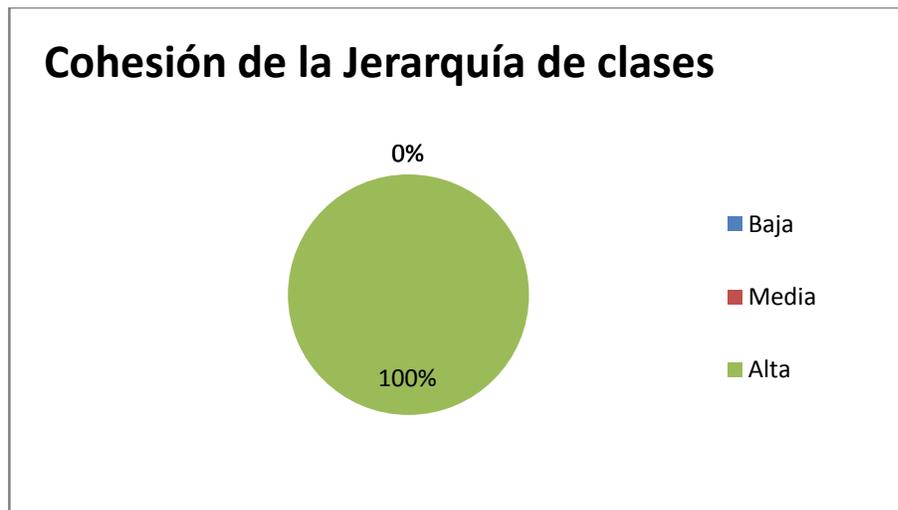


Figura 29. Representación de la incidencia de los resultados de la evaluación de la métrica ND en el atributo Cohesión de la Jerarquía de clases.

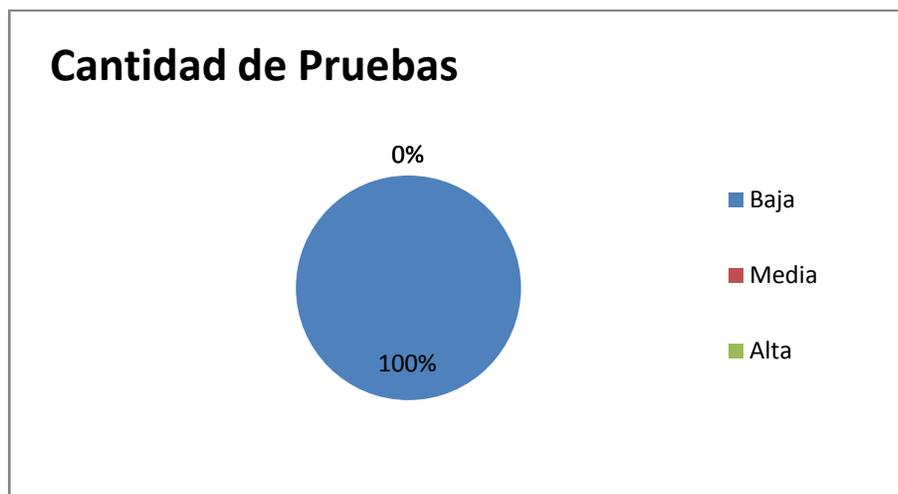


Figura 30. Representación de la incidencia de los resultados de la evaluación de la métrica ND en el atributo Cantidad de Pruebas.

Haciendo un análisis de los resultados obtenidos en la evaluación del instrumento de medición de la métrica ND, se puede concluir que el diseño del componente Carga Inicial tiene una calidad aceptable. A no ser en el atributo Reutilización ya que al no existir herencia ninguna de las clases posee descendientes y sus resultados no son favorables en un 100 %. En cuanto al atributo Abstracción de la clase base, se

muestra como existe una tendencia a la conservación de la abstracción. Tanto para la Cohesión de la jerarquía de clases como para el atributo Cantidad de Pruebas los índices son positivos favoreciendo esto al diseño.

Resultados del instrumento de evaluación de la métrica Número de Operaciones Redefinidas (NOR).

Ver instrumentos y tabla de resultados en el Anexo 5: Instrumento de medición de la métrica de Número de Operaciones Redefinidas (NOR).



Figura 31. Representación en % de los resultados obtenidos en el instrumento agrupados en los valores existentes.



Figura 32. Representación de la incidencia de los resultados de la evaluación de la métrica NOR en el atributo Complejidad del Mantenimiento.

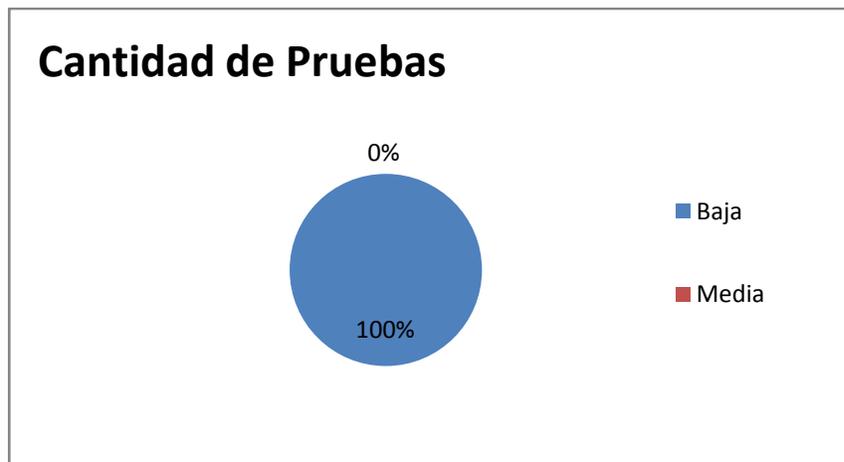


Figura 33. Representación de la incidencia de los resultados de la evaluación de la métrica NOR en el atributo Cantidad de Pruebas.

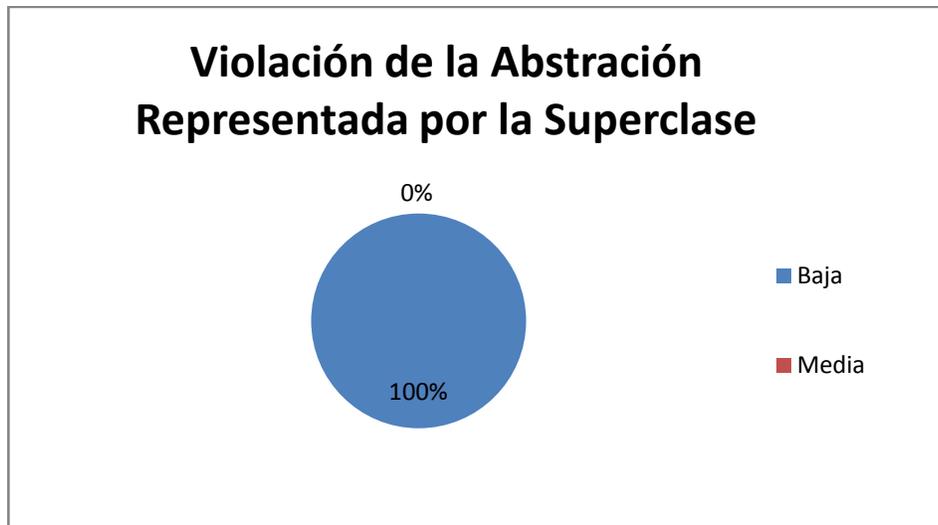


Figura 34. Representación de la incidencia de los resultados de la evaluación de la métrica NOR en el atributo Violación de la Abstracción representada por la superclase.

Haciendo un análisis de los resultados obtenidos en la evaluación del instrumento de medición de la métrica NOR, se puede concluir que el diseño del componente Carga Inicial tiene una calidad aceptable teniendo en cuenta que ninguna de las clases incluidas en el análisis redefine algún método heredado. Además, los indicadores se comportan de forma adecuada para los atributos de calidad Complejidad del Mantenimiento, Cantidad de Pruebas, Violación de la Astricción representada por la superclase.

3.2 Matriz de inferencia de indicadores de calidad

La matriz de inferencia de indicadores de calidad permite conocer si el resultado obtenido de la relación entre los atributos y las métricas asociadas a ellos para el componente, es positivo o negativo. Es una representación estructurada de las métricas y los atributos utilizados para evaluar la calidad de la solución propuesta. Aplicando una escala numérica a estos resultados (donde 1 corresponde a positivo, 0 a negativo y un signo de menos (-) si este es nulo) si no existe relación y realizando un cálculo donde se promedia la suma de los valores obtenidos de un atributo por cada métrica y dividiendo el resultado por la cantidad de métricas evaluadas¹⁷, se pueden clasificar los atributos en buenos, regulares y malos (ver Tabla 25).

Atributos/Métricas	TOC	RC	PH	ND	NOR	Promedio
Responsabilidad	1	-	-	-	-	1
Complejidad del diseño	-	-	1	-	-	1
Complejidad de implementación	1	-	-	-	-	1
Reutilización	1	1	0	0	-	0.5
Acoplamiento	-	1	-	-	-	1
Complejidad del mantenimiento	-	1	-	-	1	1
Cantidad de pruebas	-	1	-	1	1	1
Nivel de cohesión	-	-	-	1	-	1
Abstracción del diseño	-	-	-	1	1	1

Tabla 25. Matriz de inferencia de indicadores de calidad.

Categoría	Criterio
Malo	≤ 0.4

¹⁷ La relación nula (-) no se promedia.

Regular	>0.4 y ≤ 0.7
Bueno	>0.7

Tabla 26. Rango de valores para la evaluación técnica de los atributos de calidad evaluados por cada métrica.

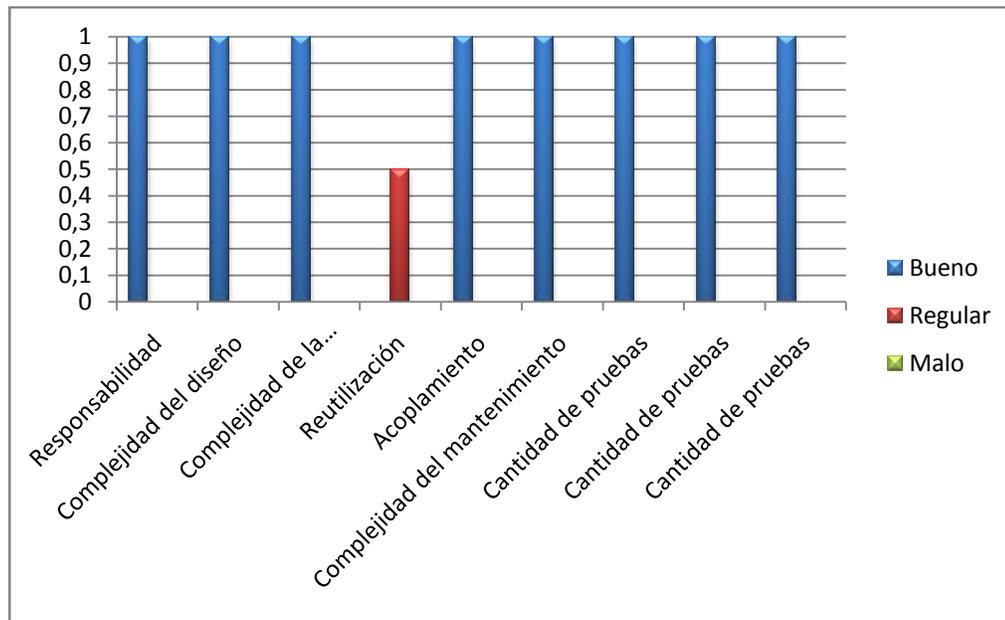


Figura 35. Gráfica de los resultados obtenidos de los atributos de calidad evaluados en las métricas.

Analizando la gráfica generada a partir de la matriz de inferencia de indicadores de calidad (ver Tabla 25) y el rango de valores para la evaluación técnica de los atributos de calidad evaluados por cada métrica (ver Tabla 26), se puede valorar que la solución propuesta mantiene una factible funcionalidad y valores positivos para cada uno de los atributos de calidad relacionados con las métricas evaluadas.

Conclusiones

Se abordaron las métricas de software, orientadas a objetos y clases, permitiendo definir cuales aplicar a la solución. Basándose en los atributos de calidad trazados se pudo realizar una medición del componente de Carga Inicial, la cual arrojó como resultado que la solución se puede valorar de aceptable.

CONCLUSIONES GENERALES

A manera de conclusión se puede reflejar que los estudios realizados sobre los flujos de trabajo arrojaron que estos se han convertido en una solución factible para la mayoría las actividades que involucren procesos, además de brindar un ahorro de tiempo y mejora en la calidad para las entidades que utilicen sistemas de este tipo.

Tomando como base el planteamiento anterior y permitiendo darle cumplimiento al primer objetivo específico, se realizó un estudio sobre el estado del arte de la teoría de flujos de trabajo, donde se refleja la fuerte base matemática que tienen los mismos y la necesidad de utilizar un modelo de referencia de soluciones genéricas, para el desarrollo de un sistema que gestionen este tipo de flujos. Además, se concibió la forma mediante la cual se desarrolló la solución propuesta.

Enmarcados en el segundo capítulo del trabajo, se realizó un levantamiento de requisitos de la solución, así como un diseño de la misma, generando el diagrama de clases y artefactos de implementación tales como el modelo de datos y el diagrama de componentes, por último se desarrollaron los casos de prueba que permitieron identificar y comunicar las condiciones que se llevaron a cabo durante las pruebas. Permitiendo de esta forma darle cumplimiento al segundo y tercer objetivo específico.

La modelación realizada facilitó la implementación de la solución obteniéndose una mayor calidad y reutilización, por lo que queda cumplido el cuarto objetivo específico.

Por ultimo se efectuó una validación del diseño mediante la utilización de instrumentos de medición que se inspiraron en métricas para la calidad. Los resultados arrojados permitieron concluir que el diseño presentaba valores positivos en indicadores de calidad tales como Reutilización, Facilidad de Mantenimiento, Complejidad del Diseño, Complejidad de Implementación, Cohesión, Acoplamiento, Cantidad de pruebas entre otros. Esto apoya la afirmación de que el diseño desarrollado se puede considerar como aceptable.

RECOMENDACIONES

A manera de recomendación se incluyen los siguientes aspectos:

- Desarrollar un diseñador de procesos con interfaz gráfica amigable para brindarle al usuario la posibilidad de crear y configurar procesos abstrayéndolo de la BD.
- Incluir en versiones posteriores el uso de estándares que garanticen una eficiente interoperabilidad entre sistemas de flujos de trabajo.

A manera de recomendación general se sugiere que sea fomentado el estudio y uso de sistemas basados en flujos de trabajo, para poder incluirlos en varios proyectos productivos de la UCI, beneficiándose éstos de las ventajas que brindan los sistemas de este tipo, entre las cuales se encuentran ahorro de tiempo y aumento de la calidad.

BIBLIOGRAFÍA

Aalst, W.M.P. van der. *Workflow Patterns*.

Bergmann, Sebastian. *Design and Implementation of a Workflow Engine*. s.l. : Rheinische Friedrich-Wilhelms-Universität. Tesis de Diploma. 1247261.

Canseco, V. and Gerónimo, G. *Introducción a los Sistemas Colaborativos: Groupware & Workflow*. Universidad Tecnológica de Mixteca. Oaxaca, Mexico : s.n.

Carlsen, S. 1997. *Conceptual Modeling and Composition of Flexible Workflow Models*. s.l. : Norwegian University of Science and Technology, 1997.

Cervantes Canales, Jair. 2005. *Representación y aprendizaje de conocimiento con redes de Petri difusas*. México, D. F. : Centro de Investigación y de Estudios Avanzados, 2005.

Chidamber and Kemerer. *Management Use of Metrics for Object-Oriented Software: An Exploratory Analysis*. s.l. : IEEE Trans. Software Engineering.

DAML. The DARPA Agent Markup Language Homepage. [Online] <http://www.daml.org/>.

Equipo Arquitectura del ERP Cedrux. 2008. *Especificación Técnica para el marco de la arquitectura*. 2008.

ERP-Cuba. 2008. *Documento Visión*. Universidad de las Ciencias Informáticas. Ciudad de La Habana : s.n., 2008. Rector.

—. **2008.** *Modelo de Desarrollo*. Universidad de las Ciencias Informáticas. Ciudad de La Habana : s.n., 2008.

Fernandes, , Sergio Miguel, Cachopo, Joao and Rito Silva, Antonio. *Supporting Evolution in Workflow Definitions Languages*. INESC-ID/Technical University of Lisbon.

García Mato, Lic. Rosa María. 1999. *Diseño de bases de datos*. 1999.

González, Jesús. 2006. *Sistemas workflow: funcionamiento y metodología de implantación*. España : s.n., 2006. 84-9704-219-0.

- Hollingsworth, David.** The Workflow Reference Model. Workflow Management Coalition. [Online] <http://www.wfmc.org/standards/docs/tc003v11.pdf>.
- IBM.** Business Process Execution Language for Web Services version 1.1. [Online] <http://www.ibm.com/developerworks/library/specification/ws-bpel/>.
- Lazo, Ochoa and Yzquierdo, Raykenler. 2007.** *El modelo de diseño del sistema HyperWeb. Módulos de Tratamiento Farmacológico y Configuración.* Habana : s.n., 2007.
- Leymann, Prof. Dr. Frank. 2001.** *Web Services Flow Language(WSFL 1.0).* IBM Academy of Technology, IBM Software Group, IBM. 2001.
- Lorenz, M. and Kidd, J. 1994.** *Object Oriented Metrics.* s.l. : Englewood, 1994.
- MANOLESCU, DRAGOS-ANTON.** *MICRO-WORKFLOW: A WORKFLOW ARCHITECTURE SUPPORTING.* Diploma de Inginer, Universitatea Politehnica Bucuresti, 1995.
- National Institute of Standards and Technology.** National Institute of Standards and Technology. [Online] <http://www.nist.gov/index.html>.
- Object Management Group.** OMG Object Management Group. [Online] IBM, HP, Sun Microsystems, Apple Computers. <http://www.omg.org/>.
- OMG.** *Workflow Management Facility Specification.*
- Pressman, S. 2002.** *Ingeniería de Software: Un enfoque Práctico.* Quinta edición. 2002.
- Prof. Dr. Caro, José L.** *Tecnología Workflow: estado actual de la investigación.* Dpto. Lenguajes y Ciencias de la Computación, Grupo de Investigación SICUMA, Sistemas de Información Cooperativos de la UMA, Universidad de Málaga.
- R. de Soto, Adolfo and Cuervo Fernández, Eva. 2006.** *Nuevas Tendencias en Sistemas de Información: Procesos y Servicios.* 2006. pp. 129-158.
- Ramos, Germán, Regourd, Jorge and Rivademar, Juan J.** *Tesis de graduación de la Licenciatura en Informática.* Facultad de Ciencias Exactas, Universidad Nacional de La Plata. La Plata : s.n. Tesis.
- Ravi Khemuka, Atul. 2003.** *Workflow Modeling Using Finite Automata.* s.l. : Department of Industrial and Management Engineering, College of Engineering, University of South Florida, 2003.

Reutilización de Requisitos de usuario. . **García, Peñalvo. Francisco José and López, Oscar. 2002.** 2002, Revista Comlombiana de Computación .

Saffirio, Mario. Tecnologías de Información y Arquitectura de Sistemas. [Online] [Cited: Enero 8, 2008.] <http://msaffirio.wordpress.com/2007/10/27/workflow/>.

The application of Petri nets to workflow management. **Aalst, W.M.P. van der. 1998.** 1, 1998, The Journal of Circuits, Systems and Computers., Vol. 8, pp. 21-66.

Universidad de Guadalajara. 2008. *Métricas, Estimación y Planificación en Proyectos de Software.* Universidad de Guadalajara. Guadalajara : s.n., 2008.

van derAalst, W. M. P. 1996. *Petri-net-based Workflow Management Software.* 1996. Proceedings of the NFSWorkshop onWorkflow and Process Automation in Information Systems.

Workflow Management Coalition. WfMC. [Online] [Cited: Enero 8, 2009.] <http://www.wfmc.org/>.

Wortmann, Hans and Szirbik, Nick. *ERP and Workflow Systems.* University of Groningen, Faculty of Management and Organisation, Information SystemsCluster, PoBox 800, 9700 AV Groningen, The Netherlands.

GLOSARIO DE TÉRMINOS

Arquitectura de software: Básicamente indica la estructura, funcionamiento e interacción entre las partes del software.

BSD: Licencia de software libre resultado del proyecto llamado *Berkeley Software Distribution* en la universidad del mismo nombre en California en la década de los 70s. La *Free Software Foundation* la considera compatible con la GPL.

CORBA: *Common Object Request Broker Architecture* es un estándar que establece una plataforma de desarrollo de sistemas distribuidos facilitando la invocación de métodos remotos bajo un paradigma orientado a objetos, definido y controlado por el OMG.

Flujo de trabajo primitivo: Es aquel que no es definido en función de otro flujo de trabajo, de forma que es un flujo de trabajo que debe ser ejecutado por un agente (humano o no), sin necesidad de más definición.

Flujo de trabajo: Conocido internacionalmente como workflow, es la automatización de un proceso de negocio, total o parcial, en la cual documentos, información o tareas son pasadas de un participante a otro para efectos de su procesamiento, de acuerdo a un conjunto de reglas establecidas. (Workflow Management Coalition)

Framework: Estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado.

IBM Lotus Workflow: Es una herramienta de desarrollo de flujos de trabajo de aplicaciones independiente y fácil de usar que hereda las capacidades del flujo de trabajo nativo del software IBM Lotus Domino y acelera la creación y el desarrollo de aplicaciones orientadas a flujos de trabajo.

LiquidOffice: Solución basada en Web, para la creación, implementación y administración automática de los procesos de enrutamiento, seguimiento y aprobación de formas electrónicas.

Motor de flujos de trabajo: Software encargado de administrar y ejecutar procesos. Algunos motores de workflow pueden interactuar entre ellos.

openEDMS: Herramienta que permite gestionar gran cantidad de documentos y hacer más eficiente la creación de contenido de negocio, almacenar, revisar, distribuir y compartir la información con mayor agilidad.

Proceso: Es un conjunto de uno o más procedimientos o actividades directamente ligadas, que colectivamente realizan un objetivo del negocio normalmente dentro del contexto de una estructura organizacional que define roles funcionales y relaciones entre los mismos. (Workflow Management Coalition)

Sistema basado en flujos de trabajo: Es un sistema que define, crea y gestiona automáticamente la ejecución de modelos workflow mediante el uso de uno o más motores workflow que se encargan de interpretar la definición de procesos (mapa workflow), interactuar con los agentes y, cuando se requiera, invocar el uso de los sistemas de información implicados en el trabajo. (Workflow Management Coalition)

UML: *Unified Modeling Language* es un lenguaje de modelado de sistemas de software respaldado por el OMG.

Wf-XML: Ofrece un estándar para un motor BPM para invocar un proceso en otro motor BPM y para esperar su completamiento.

XMI: Es una especificación para el intercambio de diagramas, también conocido como XML de Intercambio de Metadatos.

XPDL: Es un formato de archivo basado en XML que puede ser usado para intercambiar modelos de procesos de negocio entre distintas herramientas.

ANEXOS

Anexo 1: Instrumento de medición de la métrica de Tamaño operacional de Clases (TOC).

	Categoría	Criterio
Responsabilidad	Baja	< =Prom.
	Media	Entre Prom. y 2* Pom.
	Alta	> 2* Prom.
Complejidad implementación	Baja	< =Prom.
	Media	Entre Prom. y 2* Pom.
	Alta	> 2* Prom.
Reutilización	Baja	> 2*Prom.
	Media	Entre Prom. y 2* Pom.
	Alta	<= Prom.

Tabla 27. Rango de valores para la evaluación técnica de los atributos de calidad (Responsabilidad, Complejidad de implementación y Reutilización) relacionados con la métrica TOC.

Clase	Cantidad de Procedimientos	Responsabilidad	Complejidad	Reutilización
DatWorkflowModel	9	Media	Media	Media
DatChequeoModel	4	Baja	Baja	Alta

Tabla 28. Resultados de la evaluación de la métrica TOC y su influencia en los atributos de calidad (Responsabilidad, Complejidad de Implementación y Reutilización).

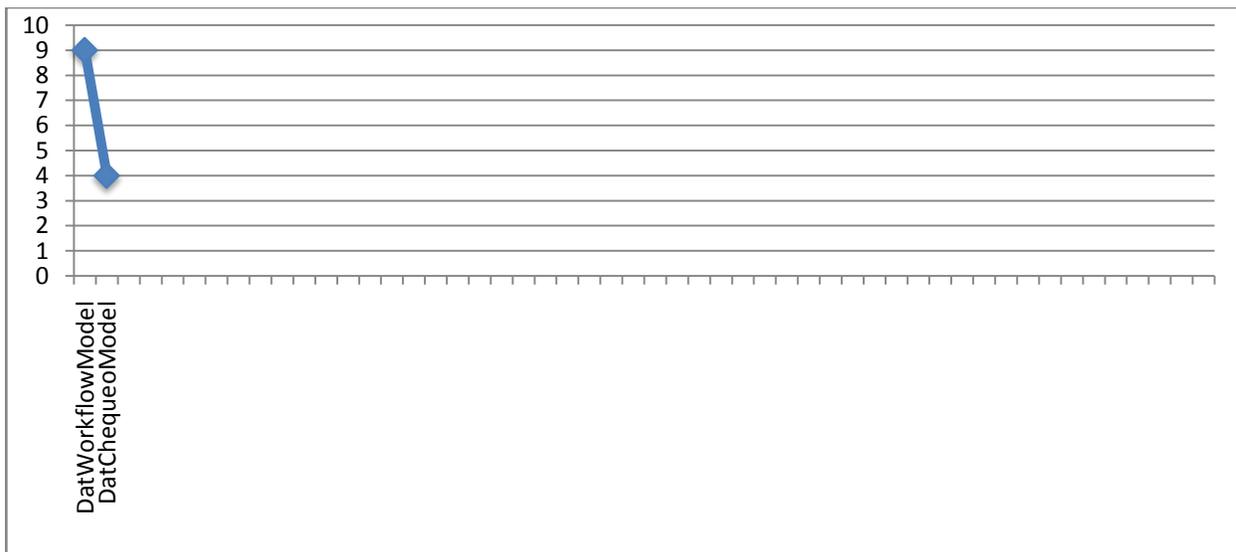


Figura 36. Gráfica de los resultados de la evaluación de la métrica TC y su influencia en los atributos de calidad (Responsabilidad, Complejidad de Implementación y Reutilización).

Anexo 2: Instrumento de medición de la métrica de Relaciones entre Clases (RC).

	Categoría	Criterio
Acoplamiento	Ninguno	0
	Bajo	1
	Medio	2
	Alto	>2

	Categoría	Criterio
Complejidad Mant.	Baja	\leq Prom.
	Media	Entre Prom. y 2*Prom.
	Alta	$> 2*$ Prom.

	Categoría	Criterio
Reutilización	Baja	$>2*$ Prom.
	Media	Entre Prom. y 2*Prom.
	Alta	\leq Prom.

	Categoría	Criterio
Cantidad de Pruebas	Baja	\leq Prom.
	Media	Entre Prom. y 2*Prom.
	Alta	$> 2*$ Prom.

Tabla 29. Rango de valores de para la evaluación técnica de los atributos de calidad (Acoplamiento, Complejidad de Mantenimiento, Reutilización y Cantidad de Pruebas) relacionados con la métrica RC.

Clase	Cantidad de Relaciones de Uso	Acoplamiento	Complejidad Mant.	Reutilización	Cantidad de Pruebas
DatWorkflowModel	1	Medio	Media	Media	Media
DatChequeoModel	0	Bajo	Baja	Alta	Baja

Tabla 30. Resultados de la evaluación de la métrica RC y su influencia en los atributos de calidad (Acoplamiento, Complejidad de Mantenimiento, Reutilización y Cantidad de Pruebas).

Anexo 3: Instrumento de medición de la métrica de Profundidad de Herencia (PH).

Clase	Clase padre	Niveles del árbol de herencia
DatWorkflowModel	-	0
DatChequeoModel	-	0

Tabla 31. Resultados de la evaluación de la métrica PH.

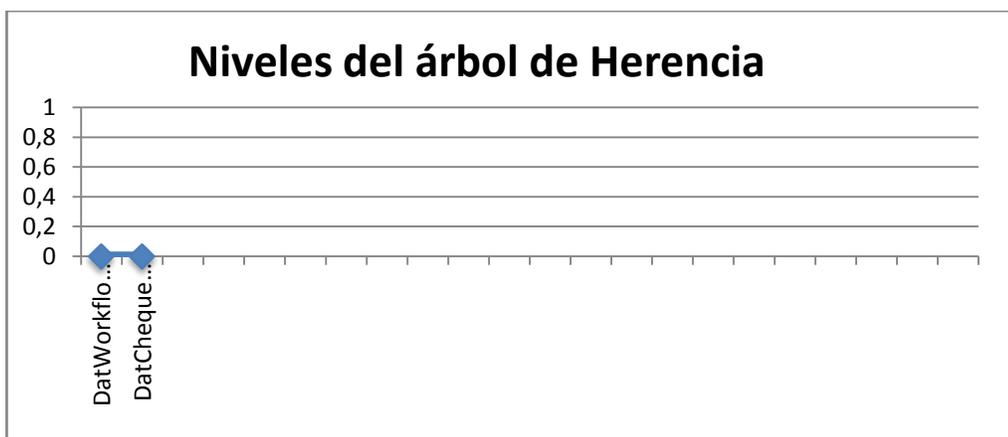


Figura 37. Gráfica de los resultados de la evaluación de la métrica PH.

Anexo 4: Instrumento de medición de la métrica de Número de Descendientes (ND).

	Categoría	Criterio
Reutilización	Baja	\leq Prom.
	Media	Entre Prom. y $2 \times$ Prom.
	Alta	$> 2 \times$ Prom.

	Categoría	Criterio
Abstracción	Indefinida	> 5
	Afectada	Entre 2 y 5
	Definida	≤ 2

	Categoría	Criterio
Cohesión	Baja	> 5
	Media	Entre 2 y 5
	Alta	≤ 2

	Categoría	Criterio
Cantidad de Pruebas	Baja	≤ 2
	Media	Entre 2 y 5
	Alta	> 5

Tabla 32. Rango de valores de para la evaluación técnica de los atributos de calidad (Reutilización, Abstracción del diseño, Nivel de Cohesión y Cantidad de Pruebas) relacionados con la métrica ND.

Clase	Clase padre	Número de descendientes	Reutilización	Abstracción de la clase base	Cohesión de la JC	Cantidad de Pruebas
DatWorkflowModel	-	0	Baja	Definida	Alta	Baja
DatChequeoModel	-	0	Baja	Definida	Alta	Baja

Tabla 33. Resultados de la evaluación de la métrica ND y su influencia en los atributos de calidad (Reutilización, Abstracción del diseño, Nivel de Cohesión y Cantidad de Pruebas).

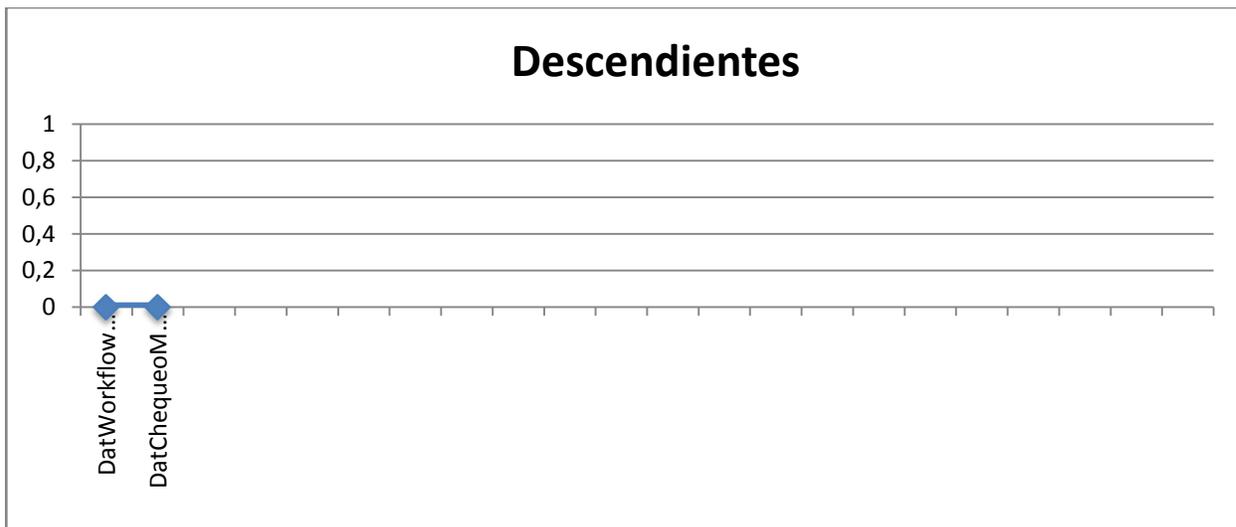


Figura 38. Representación de los resultados obtenidos al aplicar los instrumentos que evalúan la métrica ND.

Anexo 5: Instrumento de medición de la métrica de Número de Operaciones Redefinidas (NOR).

	Categoría	Criterio
Complejidad de Mantenimiento	Baja	0
	Media	Entre 0 y 2
Cantidad de Pruebas	Baja	0
	Media	Entre 0 y 2
Violación de la Abstracción representada por la superclase	Baja	0
	Media	Entre 0 y 2

Tabla 34. Rango de valores de para la evaluación técnica de los atributos de calidad (Abstracción del diseño, Cantidad de pruebas, Complejidad de Mantenimiento) relacionados con la métrica NOR.

Clase	Clase padre	No de Operaciones redefinidas	Calidad del Diseño	Complejidad de Mant.	Cantidad de Pruebas	Violación de ARS
DatWorkflowModel	-	0	Buena	Baja	Baja	Baja
DatChequeoModel	-	0	Buena	Baja	Baja	Baja

Tabla 35. Resultados de la evaluación de la métrica NOR y su influencia en los atributos de calidad (Abstracción del diseño, Cantidad de pruebas, Complejidad de Mantenimiento).

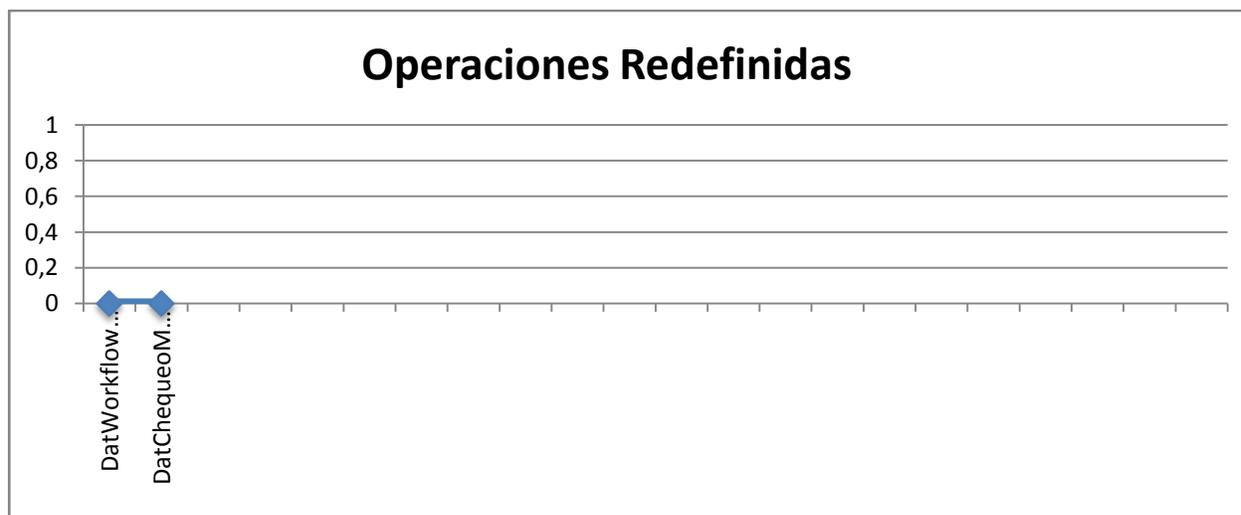


Figura 39. Representación de los resultados obtenidos al aplicar los instrumentos que evalúan la métrica NOR.

Anexo 6: Direcciones de ejecución y chequeo de las tareas.

Ejecución de la Tarea	Chequeo de la Tarea
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenNivelJerarquico
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenOrganos
metadatos/index.php/estructura/gestionarestructura	metadatos->ExistenEstructuras
seguridad/index.php/gestnomdominio/gestnomdominio	seguridad->DominioModificado
seguridad/index.php/gestrol/gestrol	seguridad->existeMasUnRol
seguridad/index.php/gestusuario/gestusuario	seguridad->existeUsuarioRol
configuracion/ejercicios/index.php/ejercicios/gestionarejercicios	parametros->HayEjercicioyPeriodo
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfContabilidad
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfCostosProcesos
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfPlanificacion

configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfCapitalHumano
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfInventario
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfActivosFijos
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfFacturacion
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfCobroPago
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfConfiguracion
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfMultimoneda
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfAuditoria
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfEstructurayComposicion
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfBanco
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfCaja
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfUsuario
configuracion/fecha/index.php/gestionarfecha/gestionarfecha	parametros->TieneFechaConfPortal
configuracion/subsistemas/index.php/gestionarentidad/gestionarentidad	subsistemas->SubsistemadeCierreContabilidad
configuracion/subsistemas/index.php/gestionarentidad/gestionarentidad	subsistemas->SubsistemadeCierreCostosProcesos
configuracion/subsistemas/index.php/gestionarentidad/gestionarentidad	subsistemas->SubsistemadeCierreCapitalHumano
configuracion/subsistemas/index.php/gestionarentidad/gestionarentidad	subsistemas->SubsistemadeCierreInventario
configuracion/subsistemas/index.php/gestionarentidad/gestionarentidad	subsistemas->SubsistemadeCierreActivosFijos
configuracion/subsistemas/index.php/gestionarentidad/gestionarentidad	subsistemas->SubsistemadeCierreFacturacion
configuracion/subsistemas/index.php/gestionarentidad/gestionarentidad	subsistemas->SubsistemadeCierreCobroPago
configuracion/subsistemas/index.php/gestionarentidad/gestionarentidad	subsistemas->SubsistemadeCierreMultimoneda
configuracion/subsistemas/index.php/gestionarentidad/gestionarentidad	subsistemas->SubsistemadeCierreBanco

configuracion/subsistemas/index.php/gestionarentidad/gestionarentidad	subsistemas->SubsistemadeCierreCaja
configuracion/multimoneda/index.php/gestmultimoneda/gestmonedacontable	parametros->TieneMonedas
configuracion/multimoneda/index.php/gestmultimoneda/gestmonedasentidad	parametros->TieneMonedaContable
seguridad/index.php/gestrol/gestrol	seguridad->existeMasUnRol
seguridad/index.php/gestusuario/gestusuario	seguridad->existeUsuarioRol
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenGrupoComplejidad
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenTipoEscalaSalarial
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenNivelUtilizacion
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExisteTipoPlantilla
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenTipoCalificador
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenCategoriaOcupacional
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenCatCivil
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenTipoCifra
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenEscalaSalarial
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenCalificador
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExisteTipoPlantilla
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenGradoMilitar
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenSubCategorias
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenCargocivil
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenCargoMilitar
metadatos/index.php/nomenclador/gestionarnomenclador	metadatos->ExistenTecnicas
metadatos/index.php/estructura/gestionarestructura	metadatos->ExistenMedios
contabilidad/contabilidad_financiera/configuraciones/index.php/estructuraeconomica/estructuraeconomica	contabilidad->ValidarTareaEstructura

contabilidad/contabilidad_financiera/configuraciones/index.php/clasificadorgrupo/clasificadorgrupo	contabilidad->ValidarTareaGrupo
contabilidad/contabilidad_financiera/configuraciones/index.php/clasificadorcontenidoeconomico/clasificadorcontenidoeconomico	contabilidad->ValidarTareaContenido
contabilidad/contabilidad_financiera/nomenclador_cuentas/index.php/configurarnomapertura/configurarnomapertura	contabilidad->ValidarTareaConfAperturas
contabilidad/contabilidad_financiera/nomenclador_cuentas/index.php/clasificadorcuentas/clasificadorcuentas	contabilidad->ValidarTareaCuentas
contabilidad/contabilidad_financiera/configuraciones/index.php/clasificadorcuentasconf/clasificadorcuentasconf	contabilidad->ValidarTareaCuentaRedondeo
contabilidad/contabilidad_financiera/configuraciones/index.php/estadosfinanciero/estadosfinanciero	contabilidad->ValidarTareaEstadof
costos_procesos/nom_conf/index.php/opnomenclador/opnomenclador	costosprocesos->hayCentros
costos_procesos/nom_conf/index.php/opnomenclador/opnomenclador	costosprocesos->hayElementos
costos_procesos/nom_conf/index.php/opcuentasgastos/opcuentasgastos	costosprocesos->hayCuentas
costos_procesos/nom_conf/index.php/opasociacion/opasociacion	costosprocesos->hayAsociaciones
contabilidad/contabilidad_financiera/comprobante_operaciones/index.php/comprobante/comprobante	contabilidad->ValidarTareaComprobante
capital_humano/persona/index.php/gestionarpersona/gestionarpersona	capitalhumano->ExistenPersonas
capital_humano/pagos_adicionales/index.php/Gestionarpagosadicionales/gestionarpagosadicionales	pagos_adicionales->ExistenPagosAdicionales
capital_humano/puesto_trabajo/index.php/Gestionarplantillapuestotrabajo/gestionarplantillapuestotrabajo	puesto_trabajo->ExistenPlantillaPuestoTrabajo
capital_humano/puesto_trabajo/index.php/Gestionarpuestotrabajo/gestionarpuestotrabajo	puesto_trabajo->ExistenPuestoTrabajo
capital_humano/incidencias/index.php/gestimpuesto/gestimpuesto	incidencias->ExistenImpuestos
capital_humano/submayores/index.php/gestionartiporetencion/gestionartiporetencion	submayores->ExistenTiposRetenciones
capital_humano/submayores/index.php/gestionarprograma/gestionarprograma	submayores->ExistenProgramas
capital_humano/nomina/index.php/gestionarperiodopago/gestionarperiodopago	nomina->ExistenPeriodoPago
capital_humano/trabajador/index.php/Emitirmovimientoalta/emitirmovimientoalta	trabajador->ExistenMovAlta
capital_humano/incidencias/index.php/gestincidencia/gestincidencia	incidencias->ExistenTipoIncidencia

capital_humano/incidencias/index.php/asociarimpinc/asociarimpinc	incidencias->ExistenIncidenciaImpuesto
capital_humano/submayores/index.php/gestionarsubmayorretenciones/gestionarsubmayorretenciones	submayores->ExistenRegistroRetenciones
capital_humano/submayores/index.php/gestionarsubmayorvacaciones/gestionarsubmayorvacaciones	submayores->ExistenSubmayorVac
capital_humano/nomina/index.php/gestionartipoajuste/gestionartipoajuste	nomina->ExistenTipoAjuste
configuracion/comprobantetipo/gestioncomprobante/index.php/gestionaroperacionespentidad/gestionaroperacionespentidad	parametros->HayReglasContables