

Universidad de las Ciencias Informáticas

Facultad 3



**Trabajo de Diploma para optar por el título de
Ingenieros en Ciencias Informáticas**

Título: Desarrollo de un componente para la realización de análisis financieros en CedruX.

Autores:

Miladis Tur Zamora

Juan Carlos Domínguez Gómez

Tutores:

Msc. Joisel Pérez Pérez

Ing. Mónica Díaz Pena

La Habana, Junio 2013. "Año 55 de la Revolución"



“Pudiéramos decir que en todas aquellas áreas para el desarrollo científico y técnico, nuestro país ha ido creando favorables condiciones y ha ido ocupando lugares destacados.”

“Sólo la ciencia, la técnica y la productividad por hectárea podrán enfrentar el grandioso desafío que tiene por delante un planeta que se empobrece y cuya tierra agrícola y agua potable disminuyen año por año”.

Líder indiscutible de la Revolución Cubana

Comandante en Jefe

Fidel Castro Ruz

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Facultad 3 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Juan Carlos Domínguez Gómez

Miladis Tur Zamora

Joisel Pérez Pérez

Mónica Díaz Pena

Msc. Joisel Pérez Pérez:

Máster en Informática Aplicada, Universidad de las Ciencias Informáticas.

Categoría Docente: Instructor.

Años de graduado: 5.

Arquitecto de datos de la línea Contabilidad del proyecto ERP Cuba.

Ing. Mónica Díaz Pena:

Ingeniero en Ciencias Informáticas, Universidad de las Ciencias Informáticas.

Categoría Docente: Adiestrada.

Años de graduado: 3.

Desarrolladora de la línea Contabilidad del proyecto ERP Cuba.

Juan Carlos

La esencia de vivir es ir hacia adelante. La vida, en realidad, es una calle de sentido único en la cual mientras avanzamos nos encontramos con infinitudes de obstáculos que debemos superar, para lograr convertirnos en personas de bien. En muchas ocasiones, la ayuda de familiares, amigos y compañeros nos es imprescindible para lograr vencer esas dificultades, poder llegar a la meta propuesta y hacer realidad nuestros sueños. Aunque los agradecimientos no sean suficientes para expresar lo que en realidad sentimos por esas personas, en este día tan importante de mi vida quiero agradecer a:

- *Mi mamá por ser la persona más importante de mi vida. Gracias por todo lo que me has dado sin condición alguna, por estar junto a mí cada vez que lo he necesitado, por todo el apoyo que me ha brindado, por soportarme, creer y confiar en mí. Este triunfo también es suyo y lo que soy hoy en gran parte te lo debo a ti, y aunque estas palabras sean pequeñas quiero que sepas que te quiero con todo mi corazón, mucho más de lo que alguien pueda imaginar.*
- *Mi papá por apoyarme en tantos momentos, aunque muchas veces hemos estado lejos, sé que siempre me tiene en sus pensamientos. A ti te debo en gran medida este logro, este sueño, gracias por hacer todo lo que hiciste para que llegara hasta aquí, he aprendido muchas cosas de ti, y te he mantenido presente en varias situaciones de mi vida. Te quiero mucho papi, infinitamente.*
- *Mi familia de Holguín, a mi abuela, mi abuelo en donde sea que se encuentre, a mi primo Antonio y a mi prima Luz, a mis tías Deysi, Digna y Dalia y a mi tío Romilio por brindarme su ayuda en todo momento.*
- *Mi familia de Cacocum, a mis hermanos Yoannis y Yoan, a mi madrastra Gisela, y a mi otra abuela Herminda, a mis tíos y tías, a mi prima Haydee y a "Julin", y a todos mis primos y primas, amigos y amigas, que aunque no los menciones porque son muchos, siempre los tengo presente.*
- *Mi novia Ania por ayudarme tanto, por brindarme tanto cariño y amor, por aguantarme en mis malos momentos y comprenderme. Eres increíblemente importante para mí y siempre te voy a amar y estaré para ti cuando así lo necesites.*
- *Mi suegra Fela y mi suegro Alcides, a mi cuñado Alcidito y a su esposa Ena, a Isabel, Eddy, Arnoldo y demás familiares y amigos de Bayamo por brindarme tanto cariño y apoyo en tan poco tiempo.*
- *Mis tutores Joisel y Mónica. A Joisel por brindarnos tanta ayuda y conocimiento en esta investigación, además de guiarnos en cada paso que fuimos dando, y ser un tutor tan dedicado con sus tesis. A Mónica por tener una solución para el código siempre que la consultamos y aclararme muchas dudas. Para los dos también va este resultado.*

- *Mis amigos de toda la vida, Anderson, Alison, Jaibar y Morejon, a mis mejores amigos de la Universidad que los considero hermanos y los tendré presente para toda la vida, Javier, Juan José, Sergio, Gustavo y Ariel, a mis compañeros del edificio y del aula, Yandy, Carlos, Roilan, David, Yasmany, Cire, Leiza, Yessika por tirarme el cabo tantas veces, en fin a todos con los que hice y mantuve una buena relación y por tantos momentos y alegrías compartidas.*
- *Mis amigos que por parte de Ania se convirtieron en buenos amigos míos, Doannis, Yoslenys, Irmel, Ricnelis, Velmour, y demás que les tengo bastante cariño.*
- *A mi compañera de tesis por soportar junto a mí cada situación engorrosa en la que nos vimos enmarcados. A mi equipo de desarrollo de la línea Contabilidad, les agradezco por todo y por acogerme entre ustedes como parte de su equipo de trabajo en los dos últimos años de la carrera.*

Miladis

Cuando llegué a esta universidad no pude imaginar todas las amistades que en ella iba a encontrar, amistades con las cuales he compartido momentos alegres y momentos para recordar toda la vida. Hoy estoy aquí a un paso de hacerme ingeniera y quiero agradecerles a todas esas personas que me han ayudado de una forma u otra durante todos estos años:

Primero que todo quiero agradecer a mi madre por ser tan fuerte como es y por resistir tantas cosas que sólo ella sabe. A mi papá, que siempre está pensando en mi bienestar.

A mis hermanos por todo su cariño incondicional.

A mi novio, mi amigo y compañero que ha estado a mi lado durante momentos difíciles ayudándome y enseñándome la mayoría de las cosas que hoy sé.

A mi familia que de una forma u otra me ha apoyado cuando la he necesitado.

A mis amistades del pre y las de mi primer año de universidad que no se olvidaron de mí, aunque tomáramos caminos diferentes en la vida.

A los amigos nuevos y viejos, por estar en las buenas y en las malas.

A todos los que un día me extendieron la mano, me dieron un consejo o aportaron un granito de arena para que hoy me hiciera ingeniera. A todos muchas gracias.

A mi tutores que supieron guiarme por el camino correcto hacia la realización completa de este trabajo de diploma.

A mi compañero de tesis por compartir conmigo todo un año de sacrificio.

A la Revolución, a Fidel y a Raúl por habernos dado la oportunidad de estudiar en este centro universitario.

A los que de una manera u otra se interesaron por el desarrollo de la tesis. Gracias

Juan Carlos

A mi madre por ser ella protagonista en gran medida de este logro, que sin ella no hubiese podido ser posible. A mi padre porque si no hubiese sido por el graduarme en esta universidad solo hubiese sido una ilusión, para él va este triunfo también.

A mi novia Ania por regalarme tantos momentos lindos y estar siempre a mi lado cuando lo necesite, y levantarme cuando me caí, a ti mi amor te dedico este triunfo.

A mis familia completa, por ambas partes, por haber hecho tanto por mí, siempre los tendré presente.

A mis amigos de la Universidad y los de mi provincia por comprender mis defectos y aun así acogerme como su hermano.

A los profesores de la línea Contabilidad y a los muchachos de cuarto año, que espero que el curso que viene tengan muchos éxitos en el desarrollo de sus tesis.

Al Comandante en Jefe Fidel Castro Ruz, por ser mi ejemplo a seguir en cada paso y acción de mi vida, brindarme la oportunidad de estudiar en esta universidad.

Miladis

A mis padres que confiaron en mí siempre, que me han amado y me han enseñado a caminar por el camino correcto, quienes dieron con amor todo cuanto pudieron para que lograra este éxito. A ese niño tan especial que siempre amaré y que espero poder ser un ejemplo para él, mi hermanito Fernando. A una persona muy especial que me ha dado su cariño, su apoyo, su ayuda en los buenos y en los malos momentos; y sobre todas las cosas nunca ha dejado de confiar en mí, mi novio Yordanis. A mis amigos por escuchar mis pensamientos, por comprenderme, por convivir con mis defectos y por quererme durante todos estos años.

RESUMEN

Entre las ramas de la economía, y desde tiempos remotos la contabilidad ha sido una herramienta para el control y seguimiento de los activos, además de que provee de información a las empresas para que estas puedan realizar el proceso de gestión, planeación y administración. El análisis financiero es imprescindible dentro de la contabilidad, porque permite mostrar las debilidades y puntos fuertes que presentan las empresas, además de su situación económica y financiera, contribuyendo así al incremento de los ingresos y la disminución de los gastos. El sistema empresarial cubano presenta la necesidad de informatizar todos sus procesos de gestión, por lo que en la Universidad de las Ciencias Informáticas se viene desarrollando una solución informática denominada CEDRUX, perteneciente al proyecto ERP, con el propósito de suplir dicha necesidad. El presente trabajo consiste en desarrollar un componente para la aplicación de técnicas de análisis financiero a dichos estados, lo cual será de gran importancia, porque facilitará la toma de decisiones en las entidades cubanas. Cuenta con un estudio del estado del arte enmarcado en los sistemas vinculados a la realización de análisis financieros. La especificación, construcción y documentación de la solución se realizó basándose en el Modelo de Desarrollo propuesto por la subdirección de producción del Centro para la Informatización y la Gestión de Entidades. Se realiza además, una descripción de las herramientas, tecnologías y lenguajes necesarios para el desarrollo de dicho componente. Se logra medir la calidad de la solución aplicando pruebas para validar la implementación de las clases.

PALABRAS CLAVES: Análisis Financiero, Técnicas de análisis financiero.

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	4
1.1. Introducción.....	4
1.2. Sistemas informatizados existentes que realizan análisis financieros	4
1.3. Modelo de desarrollo de software	6
1.4. Herramientas.....	18
1.4.1. Modelado	18
1.4.2. Herramientas de programación.....	18
1.4.3. Control de Versiones	19
1.4.4. Servidor Web	19
1.4.5. Navegador web.....	20
1.5. Marco de trabajo	20
1.6. Conclusiones parciales.....	22
CAPÍTULO 2. PROPUESTA DE SOLUCIÓN.....	23
2.1. Introducción.....	23
2.3. Propuesta del sistema	23
2.3. Modelo de negocio del proceso Realizar Análisis Financiero	24
2.4. Ingeniería de requisitos asociada al proceso Realizar Análisis Financiero....	31
2.5. Diseño de la solución.....	37
2.6. Implementación de la solución	42
2.7. Conclusiones parciales.....	44
CAPÍTULO 3. VALIDACIÓN DE LA SOLUCIÓN PROPUESTA	45
3.1. Introducción.....	45
3.2. Validación del modelo de negocio e ingeniería de requisitos	45
3.3. Validación del diseño propuesto.....	46
3.4. Validación de la aplicación	50
3.5. Conclusiones parciales.....	55
CONCLUSIONES	57
RECOMENDACIONES	58
BIBLIOGRAFÍA.....	59
GLOSARIO DE TÉRMINOS.....	65
ANEXOS	66

Anexo 1 Prototipos de interfaz	66
Anexo 2 Prototipos de interfaz de usuario funcional.....	67

Índice de Tablas

Tabla 1 Patrón Completar una Única Meta.....	11
Tabla 2 Patrón El nombre revela la intención.....	11
Tabla 3 Patrón Preciso y Legible.....	12
Tabla 4 Descripción del proceso Realizar Análisis Financiero.....	29
Tabla 5 Especificación del requisito Configurar Análisis Financiero.....	36
Tabla 6 Diccionario de datos tabla mod_contabilidad.nom_estadof.....	39
Tabla 7 Validación del Diseño.....	46
Tabla 8 Atributos de calidad que evalúa la métrica TOC.....	47
Tabla 9 Resultados al aplicar la métrica TOC.....	47
Tabla 10 Atributos de calidad que evalúa la métrica RC.....	48
Tabla 11 Resultados al aplicar la métrica RC.....	49
Tabla 12 Descripción caso de prueba Configurar Análisis Financiero.....	53
Tabla 13 Descripción de variable.....	54
Tabla 14 Juego de datos a probar.....	55

Índice de Figuras

Figura 1 Estructura del estilo arquitectónico Modelo-Vista-Controlador.....	14
Figura 2 Mapa de proceso de negocio.....	25
Figura 3 Diagrama del proceso de negocio: Realizar Análisis Financiero.....	30
Figura 4 Modelo Conceptual.....	31
Figura 5 Prototipo de IU Configurar Análisis Financiero.....	37
Figura 6 Modelo de Datos.....	38
Figura 7 Diagrama de clases Realizar Análisis Financieros.....	40
Figura 8 Aplicación del patrón Experto.....	41
Figura 9 Aplicación del patrón Creador.....	41
Figura 10 Aplicación del patrón Controlador.....	41
Figura 11 Diagrama de secuencia para Cambio en pesos y porcentajes.....	42
Figura 12 Diagrama de componente.....	43
Figura 13 Prototipo de IU funcional Configurar Análisis Financiero.....	44
Figura 14 Gráfica de la métrica TOC.....	48

Figura 15 Gráfica de la métrica RC.....	50
Figura 16 Cambio en pesos y porcentajes	66
Figura 17 Variación del Capital de trabajo	66
Figura 18 Cambios en pesos y porcentajes.....	67
Figura 19 Porcentajes componentes.	67
Figura 20 Equilibrio Financiero.....	68
Figura 21 Variación del capital de trabajo.....	68
Figura 22 Graficar Estados Financieros: Línea	69
Figura 23 Graficar Estados Financieros: Barra.....	69

INTRODUCCIÓN

Desde tiempos remotos la contabilidad ha contribuido al control y seguimiento de la información financiera en las empresas, sirviendo para conocer a ciencia cierta sobre el desempeño de sus labores comerciales en cuanto a gastos, costos e inversiones. Un elemento imprescindible dentro de esta, lo constituyen los denominados análisis financieros porque permiten mostrar las debilidades y puntos fuertes que presentan las entidades. Además, ayuda en su situación económica y financiera, contribuyendo así al incremento de los ingresos y la disminución de los gastos.

En la actualidad, debido al alto nivel de competitividad a la que se enfrentan las organizaciones a nivel mundial y al creciente desarrollo de productos informáticos, se les ha hecho indispensable a las empresas la búsqueda de mayor flexibilidad, innovación y rapidez en la toma de decisiones. Con este sentido, han surgido los Sistemas de Planificación de Recursos Empresariales (ERP, por sus siglas en inglés), que integran y automatizan muchas de las prácticas de negocio asociadas a los aspectos operativos de una empresa y logran un mejor control y gestión de los procesos que se desarrollan dentro de estas.

Cuba no está ajena a esta tendencia ya que las empresas han alcanzado un desarrollo notable debido a un grupo de transformaciones dirigidas a elevar sus resultados (Nodarse, 2007). Actualmente, muchas entidades cubanas utilizan diferentes aplicaciones importadas, las cuales no soportan todas las operaciones de gestión empresarial y no satisfacen algunas particularidades existentes en la economía cubana como es el caso de la dualidad monetaria. Por lo que al no existir en el país un sistema capaz de cubrir a cabalidad todos los requerimientos que se esperan (del Toro Ríos, et al., 2008), la Universidad de las Ciencias Informáticas (UCI), viene desarrollando un sistema desde hace varios años denominado Cedrux, perteneciente al proyecto ERP.

En la actualidad Cedrux permite obtener estados financieros a través de reportes, pero se hace difícil utilizar la información que estos muestran como base para realizar análisis financieros en el sistema. Además, no posibilitan la realización de comparaciones entre diferentes períodos definidos por el usuario. Por otro lado, se dificulta la realización de las técnicas para el análisis, porque estas técnicas se realizan a través de cálculos matemáticos sobre variaciones, porcentajes, cambios en la información y la representación de diferentes gráficos. Dichos cálculos implican

operaciones matemáticas donde se realizan multiplicaciones, divisiones, sumas y restas entre cifras que pueden llegar a ser superiores a 12 dígitos. Estos aspectos no se encuentran concebidos en el sistema, ya que solo pueden realizarse manualmente a partir de los reportes que brinda Cedrux, afectando el análisis financiero en las entidades cubanas y no satisfaciendo así las necesidades del cliente.

Dada la situación anteriormente descrita se define entonces como **problema a resolver**: la gestión de la contabilidad en Cedrux se ve afectada al no posibilitar la realización de los procesos de análisis financieros, de manera que satisfaga las necesidades del cliente en la obtención de información contable.

Partiendo del problema anteriormente planteado se ha determinado como **objeto de estudio**: las soluciones informáticas para la realización de análisis financieros. Tomando como **campo de acción**: el proceso de realización del análisis financiero en Cedrux. Definiéndose como **objetivo general de la investigación**: desarrollar un componente para la gestión de la contabilidad en Cedrux, que posibilite realizar análisis financieros de manera que satisfaga las necesidades del cliente en la obtención de información contable. Y para darle cumplimiento a este objetivo se definen los siguientes **objetivos específicos**:

- Fundamentar la investigación mediante la elaboración del Marco Teórico para sustentar los conceptos y la propuesta de desarrollo del componente.
- Realizar el análisis, diseño e implementación del componente Análisis Financiero para satisfacer las necesidades del cliente en la obtención de información contable.
- Realizar pruebas de caja negra para validar el componente implementado.

Planteando como **idea a defender**: El desarrollo de un componente para la realización de análisis financieros, posibilitará satisfacer las necesidades del cliente en la obtención de información contable para mejorar la gestión de la contabilidad en Cedrux. Como **posible resultado** se espera obtener: un componente para la realización de análisis financieros en Cedrux.

El contenido del trabajo se divide en tres capítulos:

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA.

En este capítulo se lleva a cabo el estudio del estado del arte, realizándose una investigación de varios sistemas que podrían realizar análisis financieros. Se describen

también las tecnologías y metodologías que se utilizarán en el desarrollo de las funcionalidades del componente.

CAPÍTULO 2. PROPUESTA DE SOLUCIÓN.

Este capítulo detalla las características bajo las cuales operará el sistema para el futuro componente a implementar, en correspondencia con la metodología propuesta en el capítulo 1. Se describe cómo es realizado hoy en día el negocio que se desea informatizar además de la propuesta de solución. En consecuencia, se obtendrá un conjunto de artefactos establecidos por la metodología de desarrollo escogida. Igualmente se especificará la utilización de un conjunto de patrones dentro del diseño del componente.

CAPÍTULO 3. VALIDACIÓN DE LA SOLUCIÓN PROPUESTA.

El punto de partida de este capítulo está basado en los artefactos obtenidos del análisis, el diseño y la implementación del sistema en el capítulo 2 para comenzar las diferentes validaciones y pruebas del sistema. La validación de la implementación se realiza aplicando pruebas de caja negra a partir de los diseños de casos de pruebas como parte de los artefactos que se deben generar.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1. Introducción

En este capítulo se exponen los elementos relacionados con los sistemas que permiten la realización del análisis financiero. Se realiza el estudio del estado del arte mediante el análisis de sistemas de gestión contable para comprender como se efectúa el análisis financiero. Se definen los lenguajes, las tecnologías, además de las herramientas y la metodología que se utilizan en esta investigación y en el desarrollo del componente.

1.2. Sistemas informatizados existentes que realizan análisis financieros

En la actualidad, debido al avance de la tecnología y el desarrollo de las empresas, es muy difícil controlar de forma manual su actividad contable, por lo que se han implementado sistemas que automatizan los procesos de gestión económica y financiera, conocidos como sistemas contables. Estos sistemas operan de diversas formas, pero esencialmente fomentan el desarrollo de dicha gestión, debido al gran cúmulo de información que brindan a la organización que los emplea. Un sistema contable sigue un modelo básico, ofreciendo control, flexibilidad y una relación de costo/beneficio.

Existen alrededor de todo el mundo, muchas empresas que se dedican a la creación y la comercialización de dichos sistemas. Partiendo de esto se realizó un estudio de diferentes sistemas contables para obtener información relacionada con el análisis financiero, teniendo en cuenta algunos parámetros vinculados con este como son:

- Aspectos de software y de tecnología que pueden ser empleados para el desarrollo del componente como son: librerías, ddl¹ e interfaces de usuario.
- Información sobre el proceso de realización de análisis financieros; cómo se efectúa desde el punto de vista del negocio y de la aplicación, qué cantidad de técnicas realizan y cuáles son.

Luego de una búsqueda de varios sistemas nacionales y extranjeros, se pudo encontrar uno que realiza análisis financieros a los estados financieros, tal es el caso de una **Herramienta de Microsoft Access** que permite obtener un análisis

¹ Consultar Glosario de Términos.

económico-financiero de la empresa para un mes y un ejercicio en específico, comparando los datos con los obtenidos en el mismo mes y cuatro ejercicios anteriores y realizando una previsión aproximada de lo que sucederá para el ejercicio actual completo.

Por otra parte, como se define en (ERP Access, 2009), la herramienta implementada sobre Microsoft Access para el análisis financiero, genera de forma automática toda la información que se requiere en el análisis económico-financiero, a partir de los datos obtenidos en la contabilidad. De esta forma se pueden realizar diferentes tipos de análisis como son:

- ✓ Balance de Situación (Operativo, Normal y Abreviado)
- ✓ Análisis de Balance (Comparativo y Previsional)
- ✓ Análisis de Balance Rápido
- ✓ Cuadro de Masas Patrimoniales
- ✓ Capitalización
- ✓ Garantía frente a terceros
- ✓ Fondo de Maniobra
- ✓ Test del valor Z para evaluar la situación concursal
- ✓ Razones financieras
- ✓ Estudio del Riesgo de Clientes. Presentando en un cuadro comparativo toda la distribución del crédito a clientes distinguiendo entre clientes del grupo, de moneda extranjera, deudores y clientes de dudoso cobro.
- ✓ Cuenta de Pérdidas y Ganancias (Operativa, Normal y Abreviada)
- ✓ Cuenta de Pérdidas y Ganancias Analítica (Comparativa y Provisional)
- ✓ Estudio de Resultados y Crecimiento (Comparativo y Provisional)

De todas ellas en el desarrollo del componente Análisis Financiero en Cedrux, se utilizan las concepciones brindadas para el Balance de Situación y Análisis de Balance, sirviendo de ayuda sus respectivas interfaces de usuario, las cuales brindarán una guía para realizar el diseño de algunas de las interfaces del componente. Además de dichas interfaces, también se emplean como ayuda los formatos de los reportes del análisis, que brindan una idea de cómo ofrecer un resumen del análisis de forma clara, concisa y transparente para que ayude al contador a entender la información que se muestra y contribuya a evaluar la situación financiera de la empresa. Sin embargo, en cuanto a los aspectos relacionados con la

tecnología no podrá ser reutilizado, ya que la tecnología de software sobre la cual está soportada esta aplicación no es compatible con las definidas para la construcción del componente. Por tanto, utilizando el modelo de desarrollo que se describe en el epígrafe siguiente, se realiza una solución con características propias, pero empleando como guía los aspectos que fueron encontrados en el estudio anterior.

1.3. Modelo de desarrollo de software

Para la construcción del componente, el modelo de desarrollo de software empleado es el propuesto por la Subdirección de Producción del Centro de Informatización de la Gestión de Entidades (CEIGE), y en este se describe la secuencia de actividades de alto nivel para la construcción y desarrollo de soluciones. Dicho modelo está basado en componentes y en él se muestran los diferentes artefactos que se deben generar en la medida que se construye el componente.

Desarrollo basado en componentes: Según se indica en (Colectivo de autores: UCID, 2009), un desarrollo basado en componentes lleva a alcanzar un mayor nivel de reutilización de software, aún en contextos distintos de aquellos para los que fue diseñado. Permite además que las pruebas sean ejecutadas probando cada uno de los componentes antes de probar el conjunto completo de componentes ensamblados. También al existir un débil acoplamiento entre componentes, se puede actualizar y agregar componentes según sea necesario, sin afectar otras partes del sistema. Otros de los beneficios que ofrece es que dado que un componente puede ser construido y luego mejorado continuamente, la calidad de una aplicación basada en componentes mejorará con el paso del tiempo.

Según (Obregón, 2012) por cada disciplina deben ser generados una serie de artefactos los cuales ayudan a una mejor comprensión del proceso de desarrollo que se lleva a cabo. A continuación se presentan dichos artefactos, además de otros aspectos que corresponden a cada disciplina y que son utilizados en el desarrollo del presente trabajo de diploma.

➤ **Disciplina Inicio o Estudio preliminar**

En esta disciplina se llevan a cabo las actividades relacionadas con la planeación del componente definiéndose a su vez el alcance del mismo (Obregón, 2012).

Artefactos

- Cronograma donde se listan las actividades a realizar.

➤ **Disciplina de Modelado del negocio**

En esta disciplina se realiza el modelado del negocio referente a la realización del análisis financiero, para lo que se utiliza la herramienta correspondiente al modelado definida en el epígrafe 1.4.1.

Artefactos

- Mapa de proceso de negocio.
- Diagrama del proceso de negocio.
- Modelo conceptual.
- Descripción del proceso de negocio.
- Glosario de término de la línea y del componente a implementar.

El modelado de procesos de negocio demanda la representación de los elementos que intervienen en los procesos. Por ende, es necesario utilizar un lenguaje o notación que permita modelar dichos procesos. Para ello se propone emplear la Notación para Modelado de Procesos de Negocio (BPMN, por sus siglas en inglés) que como lo indica (Moreno, 2009) “es un estándar cuyo principal objetivo es proporcionar una notación fácilmente comprensible por todos los usuarios del negocio, desde los analistas, los desarrolladores técnicos, hasta aquellos que monitorizarán y gestionarán los procesos”.

Además, como lenguaje de modelado se propone emplear el Lenguaje Unificado de Modelado (UML, por sus siglas en inglés) que como plantea (Jacobson, 2000) “es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software”.

Patrones de control de flujo candidatos a utilizar

Los patrones de control de flujo que son candidatos a utilizar en el desarrollo del trabajo son:

- **Patrón *Sequence***: Ejecuta actividades en forma de secuencia.
- **Patrón *Split***: Ejecuta actividades en paralelo.
- **Patrón *Synchronization***: Sincroniza dos hilos de ejecución paralelos.
- **Patrón *Exclusive Choise***: Selecciona un camino entre varios posibles.
- **Patrón *Simple Merge***: Unifica dos caminos alternativos sin sincronización.

➤ **Disciplina de requisitos**

En esta disciplina se realiza la ingeniería de requisitos perteneciente al proceso de negocio anteriormente modelado (Obregón, 2012).

Artefactos

- Lista de Requisitos.
- Especificación de requisitos de software.
- Prototipo de interfaz correspondiente a cada especificación de requisito.
- Criterios para validar requisitos del cliente.

Para el diseño de los prototipos de interfaz se emplea la herramienta correspondiente al modelado definida en el epígrafe 1.4.1. Por otra parte se definen las técnicas candidatas a utilizar para la elicitación y la validación de los requisitos, las cuales son:

Elicitación de requisitos: esta actividad se realiza con el fin de conocer el dominio del problema e identificar las necesidades reales de clientes y usuarios, para esto son consultadas distintas fuentes de información como clientes, usuarios y expertos en el dominio. (Toro, 2000)

Técnicas de elicitación

- **Entrevistas:** como indica (Toro, 2000) son utilizadas para obtener información sobre el negocio o sistema a automatizar y a partir de estos es que se definen los requisitos. Para realizar la entrevista se debe estudiar el dominio del problema, con el objetivo de conocer los conceptos fundamentales para el cliente y que este vea que se entienden sus planteamientos. Además, se deben seleccionar las personas a entrevistar, definir el objetivo de las entrevistas y planificarlas teniendo en cuenta adecuarse al horario de los entrevistados (Escalona, 2002).
- **Tormenta de ideas:** se basa en una reunión de pocas personas, por lo general no más de diez, donde los participantes exponen sus ideas de forma libre y espontánea. Se utiliza con el objetivo de generar ideas que brinden una visión a grandes rasgos de las necesidades del sistema, pero generalmente no sirve para obtener detalles concretos del mismo, por lo que suele aplicarse en los primeros encuentros. (Escalona, 2002)

- **Sketches y Storyboards:** comúnmente es usada por los diseñadores gráficos de aplicaciones en el entorno web. La misma consiste en representar sobre papel de forma muy bosquejada las diferentes interfaces al usuario (sketches). Estos sketches pueden ser agrupados y unidos por enlaces dando idea de la estructura de navegación (storyboard). (Escalona, 2002)
- **Observación y Análisis de Tareas:** consiste en que un observador estudia a los futuros usuarios en su ambiente de trabajo. Apunta todo aquello que es susceptible de mejora, para luego generar una serie de requisitos tentativos. El uso de esta técnica implica la visita del analista a las áreas donde se realizan los procesos. (Moreno, 2009)
- **Arqueología de documentos:** se fundamenta en determinar posibles requisitos inspeccionando la documentación utilizada por la empresa; por ejemplo, manuales de procedimientos, publicaciones, en investigaciones relacionadas con el proceso. Esta técnica sirve como complemento de las demás técnicas, a través de ella se obtiene información que de otra manera sería muy difícil de conseguir. (Moreno, 2009)
- **Sistemas Existentes:** consiste en analizar distintos sistemas ya desarrollados que estén relacionados con el proceso que se intenta informatizar. Pueden ser analizadas las interfaces de usuario, observando el tipo de información que se maneja y cómo es manejada. Esto puede ser útil para descubrir información importante a tener en cuenta, información que tal vez el cliente haya fallado en comunicar. También es útil analizar las distintas salidas que los sistemas producen (listados, consultas, reportes), porque siempre pueden surgir nuevas ideas sobre la base de estas. Esta técnica requiere cierto grado de investigación, así como contactos con otros profesionales que hayan desarrollado sistemas parecidos. (Moreno, 2009)

Técnicas a utilizar

Para la realización de este trabajo se determinó el uso de las siguientes técnicas: la denominada Tormenta de ideas, Arqueología de documentos, Entrevistas, Sistemas existentes, así como Sketches y Storyboards. Estas técnicas fueron escogidas debido a su facilidad de uso. Además el tiempo necesario para la aplicación de estas no constituye un contratiempo y mucho menos la cercanía al cliente, puesto que este trabajó conjuntamente.

Validación de requisitos: determina si los requisitos definidos son los que realmente quiere el cliente. “examina las especificaciones para asegurar que todos los requisitos del sistema han sido establecidos sin ambigüedad, sin inconsistencia, sin omisiones, que los errores detectados hayan sido corregidos y que el resultado del trabajo se ajusta a los estándares establecidos para el proceso, el proyecto y el producto”. (Pressman, 2005)

Técnicas para la validación de los requisitos

La validación de requisitos tiene como objetivo demostrar que la definición de los requisitos detalla realmente el sistema que el cliente desea. “La mayoría de las técnicas existentes consisten en revisar los modelos obtenidos en la definición de requisitos” (Moreno, 2009), seguidamente se muestran algunas de ellas:

- **Revisiones:** “es la fórmula más empleada para validación. Un grupo de personas (incluyendo usuarios) se ocupan de revisar el documento de requisitos y los modelos. Consta de tres fases: búsqueda de problemas, reunión y acuerdos”, (Moreno, 2009).
- **Prototipos:** “permite al usuario hacerse una idea de la estructura de la interfaz del sistema con el usuario, permitiendo descubrir con rapidez si el usuario se encuentra satisfecho, o no, con los requisitos”, (Moreno, 2009).
- **Auditorías:** “la revisión de la documentación con esta técnica consiste en un chequeo de los resultados contra una lista de chequeo predefinida o definida a comienzos del proceso, es decir sólo una muestra es revisada”, (Moreno, 2009).

Técnicas a utilizar

Para la validación de los requisitos se contemplaron las revisiones y prototipos, siendo esta última de mucha importancia. Estas técnicas no constituyen obstáculos en el proyecto, ya que para su ejecución, estas son proyectadas con anterioridad. Por ende, no interrumpen otras tareas que pudieran estar desarrollándose en el momento de aplicar la técnica. Además dichas técnicas son fáciles de entender y de aplicar, por lo que no instauran gasto adicional al encontrarse a disposición en los locales de trabajo del proyecto, todo lo necesario para su ejecución, contándose a su vez con la presencia de los especialistas en el tema.

Patrones utilizados

Según (Moreno, 2009), existen algunos patrones de casos de usos que pueden ser aplicados a los requisitos. En este trabajo se proponen utilizar algunos que fueron adaptados a los requisitos. Seguidamente se describen especificando el nombre, el problema y la solución según (Larman, 2001):

Nombre	Completar una Única Meta
Problema	Objetivos inadecuados pueden perder a los escritores a la hora de determinar dónde un caso de uso termina y otro comienza.
Solución	Escribir cada caso de uso dirigiéndose hacia una completa y bien definida meta.

Tabla 1 Patrón Completar una Única Meta.

Nombre	El nombre revela la intención
Problema	Los nombres genéricos sin sentido pueden no ubicar las expectativas del lector o proveer un punto focal que las personas pueden convenientemente echar para atrás. Utilizar nombres descriptivos para los casos de uso es una buena práctica, porque ellos revelan exactamente la intención de cada caso de uso.
Solución	Nombra los casos de uso utilizando un verbo activo o frase que represente la meta del actor primario.

Tabla 2 Patrón El nombre revela la intención.

Nombre	Preciso y Legible
Problema	Los casos de uso que son demasiado complicados para

	<p>lectores no técnicos, o son demasiado imprecisos para los desarrolladores, son deficientes y probablemente resulten en una deficiente construcción.</p>
Solución	<p>Escribir cada caso de uso lo suficientemente legible a fin de que los clientes se los lean, los evalúen y precisen lo suficiente a fin de que los implementadores entiendan qué están construyendo.</p> <p>Cada caso de uso que se escriba debe exactamente describir una Meta Única y Completa sin ser tan verboso que la audiencia no lo pueda leer o de tan alto nivel que no comunique la suficiente información para entenderlo adecuadamente.</p> <p>Una regla cardinal para escribirlos es “conocer la audiencia”. Determinar quién necesita esos casos de uso y escribir cada uno utilizando su terminología y un estilo que ellos puedan fácilmente entender.</p> <p>El correcto nivel de precisión y legibilidad presenta un objetivo en movimiento, así es importante que se entiendan las necesidades y habilidades de la audiencia cuando escriba cada caso de uso.</p> <p>Se debe usar un lenguaje de texto que describa el comportamiento del sistema. Incluir solo la información necesaria, pero describiendo el comportamiento lo suficientemente claro y preciso de manera que un lector sin conocimientos pueda completamente entender las consecuencias del comportamiento, sin perderse eventos importantes. Esta precisión no solo beneficia a los clientes, a los desarrolladores también, al no tener que hacer suposiciones al construir el sistema.</p>

Tabla 3 Patrón Preciso y Legible.

➤ **Disciplina Análisis y diseño**

Durante esta disciplina es modelado el sistema para que soporte todos los requisitos. Esto tributa a una arquitectura sólida y estable que se convierte en un plano para la próxima fase, ya que los artefactos que se generan en esta etapa son más formales y específicos de una implementación. (Obregón, 2012)

Artefactos

- Para definir la Arquitectura de datos el Modelo datos.
- Diagramas de clases del diseño.
- Diagramas de secuencias.

Para el modelado de artefactos mencionados se emplea la herramienta correspondiente al modelado definida en el epígrafe siguiente.

Según la definición brindada por la (IEEE, 2000) que plantea lo siguiente: “la arquitectura del software es la organización fundamental de un sistema, formada por sus componentes, las relaciones entre ellos y el contexto en el que se implantarán, y los principios que orientan su diseño y evolución”. Por tanto se aplica el tipo de arquitectura la cual está definida por el proyecto, y además está basada en el estilo arquitectónico Modelo-Vista-Controlador (MVC).

Este es un estilo arquitectónico usado principalmente en aplicaciones que manejan gran cantidad de datos y transacciones complejas, donde se requiere una mejor separación de los conceptos para que el desarrollo esté estructurado de una mejor manera, coincidiendo con (Gómez Baryolo, et al., 2012).

- **Modelo:** Esta capa es la representación específica de la información con la cual el sistema opera. Se limita a lo relativo de la vista y su controlador facilitando las presentaciones visuales complejas.
- **Vista:** Esta capa presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario.
- **Controlador:** Este responde a eventos, usualmente acciones del usuario, e invoca peticiones al modelo y notifica a la vista.

En la figura 1 se muestra la estructura del estilo arquitectónico Modelo-Vista-Controlador que utiliza Cedrux, la cual presenta la particularidad de que la vista no realiza solicitudes al modelo (Andux, 2010).



Figura 1 Estructura del estilo arquitectónico Modelo-Vista-Controlador.

En la figura anterior se puede observar que:

La vista es responsable de:

Realizar peticiones al controlador y se encarga además de la capa de presentación así como de visualizar la información mediante los navegadores los cuales contienen las interfaces de usuario.

En lo particular el controlador se encarga de:

Ejecutar las acciones sobre el modelo y además es el responsable de la capa de negocios o sea los componentes del servidor que manejan los requerimientos de http o https. El controlador también se faculta de:

- Recibir los eventos de entrada.
- Contener las reglas de gestión de eventos. Estas acciones pueden suponer peticiones al modelo o notificaciones a las vistas.

Por otra parte el modelo es el responsable de:

- Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
- Administra y gestiona el comportamiento de los datos persistentes respondiendo sobre su estado.

Patrones de diseño posibles a aplicar

Para el diseño se utilizan patrones, porque de manera general constituyen soluciones simples y elegantes a problemas específicos y comunes del diseño orientado a objetos. En este caso se propone emplear algunos de los patrones generales de

software para la asignación de responsabilidades (GRASP, por su acrónimo en inglés), ya que describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones, según lo indicado por (Larman, 1999). Dichos patrones son:

Patrón Experto: Se utiliza en diseños donde un objeto de software hace ciertas operaciones, que también son realizadas en el mundo real por el objeto que representa. Se encarga de asignar una responsabilidad al experto en información, o sea, aquella clase que cuenta con la información necesaria para cumplir la responsabilidad. Favorece el bajo acoplamiento. (Larman, 1999)

Patrón Creador: El creador guía la asignación de responsabilidades relacionadas a la creación de objetos, una tarea muy común en sistemas orientados a objetos. El intento básico de este patrón es encontrar un creador que necesite estar conectado al objeto creado en un evento en específico. Brinda soporte al bajo acoplamiento. (Larman, 1999)

Patrón Controlador: Asigna la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase. El controlador es un intermediario entre la interfaz de usuario y el núcleo de las clases donde reside la lógica de la aplicación. Este patrón sugiere que la lógica de negocios debe estar separada de la capa de presentación para aumentar la reutilización de código. (Larman, 1999)

Validación del diseño

El diseño se centra en convertir los requisitos del cliente en un modelo que al ser implementado, se obtenga el producto deseado. Habitualmente constituye el punto de partida para el desarrollo de software una vez especificados los requerimientos del mismo (Parametric Technology Corporation, 2006). Por tanto realizar una validación del mismo para verificar su calidad y flexibilidad, fortalece los cimientos para una buena implementación. En ese sentido se emplean un conjunto de métricas de software orientadas a determinar, entre otros aspectos, las características del modelo de diseño que se pueden estimar para evidenciar que el sistema será fácil de implementar en cuanto a organización. Para ello se proponen utilizar las métricas **Tamaño Operacional de Clase (TOC)** y **Relaciones entre Clases (RC)** para validar el diseño del componente Análisis Financiero en CedruX.

Tamaño Operacional de Clase

Esta métrica se establece por el número total de operaciones que están encapsuladas dentro de la clase. Si los valores de esta medida son grandes, puede implicar que la clase tenga demasiada responsabilidad, lo cual reducirá la capacidad de reutilización de la misma y complicará su implementación. Por otra parte, en cuanto menor sea el valor, más probable es que las clases tengan menos responsabilidad y complejidad, y por consiguiente más nivel de reutilización.

Relaciones entre Clases

Esta métrica se establece por la cantidad de relaciones que existen entre las clases contenidas en el diseño. Las dependencias son directamente proporcionales al nivel de acoplamiento, a la complejidad del mantenimiento y a la cantidad de pruebas a realizar sobre las clases, y es inversamente proporcional al grado de reutilización de las mismas.

➤ **Disciplina Implementación**

Luego de los resultados del análisis y diseño se implementa el sistema en términos de componentes tales como: ficheros de código fuente, scripts, ejecutables y similares.

Artefactos

- Ficheros de código.
- Prototipos de interfaz funcionales.
- Diagrama de componente.

Para la implementación del componente se emplean las herramientas correspondientes a la programación definidas en el epígrafe 1.4.2. Por otra parte para el control de versiones se utiliza la herramienta definida en el epígrafe 1.4.3. Los frameworks concebidos para la implementación, así como los lenguajes, tecnologías y notaciones, se encuentran definidos en el epígrafe 1.5, donde se puede apreciar una breve descripción de cada uno de ellos.

Con el objetivo de emitir los reportes que debe presentar el componente, se utiliza la librería **XPinter** definida por el proyecto. Su facilidad de uso para conformar los reportes con diferentes formatos, y su total compatibilidad con los que debe brindar el componente la hacen imprescindible en la implementación del mismo.

➤ **Disciplina Pruebas internas**

Durante esta disciplina se desarrollan las pruebas del grupo de calidad del centro verificando el resultado de la implementación. En ella se detectan posibles errores en la documentación y el software, es decir requisitos que el producto debería cumplir y que aún no los cumple (Obregón, 2012).

Artefactos

- Diseño de casos de prueba.
- Documento de no conformidades.
- Acta de liberación.

Los métodos de pruebas independientemente del nivel en que se enmarque la prueba, ayudan a encontrar buenos conjuntos de casos de prueba para detectar diferentes tipos de errores. Existen dos enfoques alternativos según indica (Ramírez, 2012), descritos como: pruebas de Caja Blanca y Caja Negra. En la primera mencionada se cuenta con el código, pues se conoce el mismo y se tratan de ejecutar cada uno de sus elementos. Por su parte en las pruebas de caja negra se cuenta con la interfaz y se maneja con cada uno de los elementos que la componen para llevar a cabo la validación. Para validar el componente Análisis Financiero se propone realizar pruebas de caja negra, con el objetivo de lograr la satisfacción de las necesidades del cliente.

Con el propósito de desarrollar estas pruebas se emplean de las diferentes técnicas existentes las siguientes:

- **Técnica de la Partición de Equivalencia:** se dirige a la definición de casos de prueba que descubran clases de errores, reduciendo así el número total de casos de prueba que hay que desarrollar (González, y otros, 2010).
- **Técnica del Análisis de Valores Límites:** es una técnica de diseño de casos de prueba que completa a la técnica anterior, llevando a la elección de casos de prueba en los extremos de la clase de equivalencia (González, y otros, 2010).

Niveles de prueba

A la hora de evaluar dinámicamente un sistema, se debe comenzar por los componentes más simples y pequeños e ir avanzando progresivamente hasta probar todo el software en su conjunto. Las pruebas se aplican en distintos niveles de trabajo, según lo indicado por (González, y otros, 2010), dentro de estos se distinguen:

- **Pruebas de Integración:** los componentes individuales son combinados con otros componentes para asegurar que la comunicación, enlaces y los datos compartidos ocurran apropiadamente.
- **Pruebas del Sistema:** son usualmente conducidas para asegurar que todos los módulos trabajan como sistema sin error. Es similar a la prueba de integración pero con un alcance mucho más amplio.
- **Pruebas de Aceptación:** Son realizadas principalmente por los usuarios con el apoyo del equipo del proyecto. El propósito es confirmar que el sistema está terminado, que desarrolla puntualmente las necesidades de la organización y que es aceptado por los usuarios finales.
- **Pruebas de Unidad:** conocidas también como unitarias son pruebas individuales a las unidades separadas de un sistema de software.

Los niveles que se proponen utilizar en la realización de las pruebas del componente son los correspondientes a las Pruebas de Aceptación y Pruebas de Unidad, con el objetivo de corregir los errores encontrados en el primero, antes de que el segundo nivel sea aplicado por el Departamento de Calidad del centro.

1.4. Herramientas

Para el desarrollo del componente se utilizarán varias herramientas definidas en el proyecto, que ayudarán a lograr de forma más fácil su construcción.

1.4.1. Modelado

Visual Paradigm para UML 8.0

Es una herramienta que emplea UML y permite representar el ciclo de vida completo del desarrollo de software. Posibilita la representación de varios tipos de diagramas, entre otras opciones. Dentro de sus características fundamentales se encuentran que soporta la opción de usar BPMN. (Visual Paradigm, 2007)

1.4.2. Herramientas de programación

NetBeans IDE 7.2

Este es un producto creado bajo licencias de software libre, es gratuito y no tiene restricciones de uso. Es un entorno de desarrollo integrado disponible para Windows, Mac, Linux y Solaris, de código abierto, escrito completamente en Java. Es una plataforma de aplicaciones que permite a los desarrolladores crear rápidamente

aplicaciones, utilizando un número importante de módulos para extenderlo a otros lenguajes como PHP (Zend y Symfony), JavaScript, Ajax, Groovy y Grails, y C / C + +. Entre sus características se encuentra la de poseer un sistema de proyectos basado en Ant², control de versiones y *refactoring*³. El proyecto de NetBeans está apoyado por una comunidad de desarrolladores y ofrece una amplia documentación y recursos de capacitación, según lo indicado por (Oracle Corporation, 2012).

PostgreSQL 8.3

Es un sistema de gestión de bases de datos, objeto-relacional, basado en el proyecto POSTGRES de la Universidad de Berkeley. Publicado bajo la licencia BSD⁴ según plantea (The FreeBSD Foundation, 2012). PostgreSQL como Sistema Gestor de Base de Datos garantiza la integridad de la información porque es libre y multiplataforma. Tiene amplio soporte por una comunidad mundial. Presenta características orientadas a objetos y herencia entre tablas, además de tener integridad referencial y permitir replicación asíncrona y acceso encriptado vía SSL, como plantea (Martínez, 2009).

PgAdmin III

Es una aplicación gráfica para administrar las bases de datos PostgreSQL, con licencia Open Source. Se puede usar en varios sistemas operativos como Linux, Solaris, Mac OS X y Windows. Incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor y un agente para lanzar scripts programados. La conexión al servidor puede hacerse mediante conexión TCP/IP. (Martínez, 2009)

1.4.3. Control de Versiones

Subversion 1.6.6

Es un controlador de versiones empleado en la administración de archivos utilizados en el desarrollo de software que permite seguir los cambios de los archivos y directorios a través de copias y renombrados. Presenta integración con el servidor web Apache y existen varias interfaces de Subversion, ya sean programas individuales o interfaces que lo integran en entornos de desarrollo. (CENDITEL, 2011)

1.4.4. Servidor Web

Apache 2.0

² Consultar Glosario de Términos.

³ Consultar Glosario de Términos.

⁴ Consultar Glosario de Términos.

Según (Observatorio Tecnológico, 2008) es un servidor web de software libre desarrollado por la Apache Software Foundation cuyo objetivo es servir o suministrar páginas web a los clientes web o navegadores que las solicitan. Es una tecnología gratuita y de código abierto, lo cual le brinda transparencia al software de manera que se pueda conocer lo que se está instalando como servidor. Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto. Es altamente configurable en la creación y gestión de logs⁵.

1.4.5. Navegador web

Mozilla Firefox 2.0 o superior

Es un navegador libre y de código abierto. Es usado para visualizar páginas web. Incluye corrector ortográfico, búsqueda progresiva y marcadores dinámicos. Además se pueden añadir funciones a través de complementos desarrollados por terceros. Es multiplataforma, permite la integración con el antivirus, realiza la navegación por pestañas, presenta compatibilidad para múltiples extensiones y utiliza el sistema SSL⁶ para proteger la comunicación con los servidores web, utilizando fuerte criptografía cuando se utiliza el protocolo HTTPS⁷. (Mozilla Firefox, 2009).

1.5. Marco de trabajo

Sauxe 1.5.4

El marco de trabajo Sauxe 1.5.4 posee un conjunto de componentes reutilizables que provee la estructura genérica y el comportamiento para una familia de abstracciones, logrando una mayor estandarización, flexibilidad, integración y agilidad en el proceso de desarrollo, (Baryolo, et al., 2008). Sauxe utiliza **ExtJS**⁸ para implementar la capa de presentación, se apoya en Zend-Ext, una extensión de **Zend Framework**⁹ para el desarrollo de la lógica del negocio, y para la gestión de los datos que maneja **Doctrine**¹⁰. Utiliza como estilo arquitectónico el Modelo-Vista-Controlador. También tiene como propósito insertar la inversión de controles para la integración de servicios

⁵ Consultar Glosario de Términos.

⁶ **Secure Sockets Layer**: protocolo criptográfico que proporciona comunicación segura en Internet.

⁷ **HTTPS**: (Hypertext Transfer Protocol Secure) Protocolo de transferencia de hipertexto seguro.

⁸ Consultar Glosario de Términos.

⁹ Consultar Glosario de Términos.

¹⁰ Consultar Glosario de Términos.

(Gómez Baryolo, et al., 2012). Además como herramienta de seguridad emplea el sistema **Acaxia** (Sistema de Gestión Integral de Seguridad), el cual está desarrollado sobre software libre e incorpora procesos tan importantes como la administración de conexiones y de perfiles. La estructura de estos Frameworks se encuentra formada por diferentes lenguajes, tecnologías y notaciones tales como:

- **HTML (Lenguaje de Marcas de Hipertexto)** es un conjunto de etiquetas, complementadas por extensiones que permiten dar formato a un archivo para que pueda ser visualizado en forma de página web (HTML.net, 2012).
- **XML 1.0 (Lenguaje de Etiquetado Extensible)** es un formato que permite la lectura de datos a través de diferentes aplicaciones y además es utilizado para estructurar, almacenar e intercambiar información (Colectivo editorial W3C Consortium, 2008).
- **JSON (Notación de Objetos de JavaScript)** es un formato de intercambio de datos que está constituido por una colección de pares de nombre/valor. Actualmente se ha convertido en un estándar en el desarrollo de aplicaciones web donde en ocasiones sustituye a XML para permitir la integración de servicios en el navegador del usuario como indica (Colectivo de JSON, 2011).
- **JavaScript 1.6** es un lenguaje de programación multiplataforma, interpretado y orientado a objetos lo que permite a los desarrolladores añadir y crear interactividad en el desarrollo y diseño de sitios web así como la validación de datos (Mozilla Developer Network, 2011)
- **AJAX (JavaScript y XML asíncronos)** es una técnica de desarrollo web para crear aplicaciones interactivas, la cual se ejecuta en el navegador del usuario, y mantiene comunicación asíncrona con el servidor, para de esta forma poder realizar cambios sobre la misma página sin necesidad de recargarla (ProgramacionWeb.net, 2005).
- **PHP 5.2 o posterior (Hipertext Preprocesor)** es un lenguaje de programación que se ejecuta en el servidor, gratuito e independiente de la plataforma de desarrollo, por lo que puede ser utilizado en cualquier marco de trabajo o IDE de desarrollo que lo soporte (The PHP Group, 2011).
- **PL/pgSQL** es un lenguaje de procedimientos almacenados para Postgres que está provisto por el gestor de base de datos PostgreSQL, y permite además ejecutar comandos SQL mediante un lenguaje de sentencias imperativas y uso de funciones (Martínez, 2009).

- **DQL** es el lenguaje de consulta de Doctrine que proporciona capacidades de consulta sobre los objetos del modelo (Doctrine-Project.org, 2010).

1.6. Conclusiones parciales

Una vez finalizado el presente capítulo se pudo arribar a las siguientes conclusiones:

- El análisis de los sistemas estudiados permitió obtener información relacionada con la realización del análisis financiero, tal como interfaces de usuario y formatos para mostrar la información contable, que se emplearon como guía para la implementación del componente.
- El ambiente de desarrollo integrado por el modelo utilizado y arquitectura seleccionada, además de los lenguajes, tecnologías y herramientas definidas, brindan la posibilidad de desarrollar una solución que apoye el cumplimiento de los principios de soberanía e independencia tecnológica y posibilite la realización de los análisis financieros.
- Lo anteriormente mencionado conforma la fundamentación teórica que sustenta el desarrollo de la investigación, y propicia a que el lector adquiera conocimientos acerca del tema a tratar y entienda la propuesta que a continuación se plantea.

CAPÍTULO 2. PROPUESTA DE SOLUCIÓN

2.1. Introducción

En el presente capítulo se especifican puntos importantes para llevar a cabo la implementación del componente, comenzando por la modelación del negocio así como el levantamiento de los requisitos funcionales, obteniéndose las especificaciones complementarias y los requisitos no funcionales. Por otra parte se generan los diagramas de procesos con sus respectivas descripciones textuales, lo que permite dar paso al diseño donde se realizan los diagramas de clases del diseño, diagramas de interacción y el modelo de datos. Además se especifica la utilización de un conjunto de patrones dentro del diseño del componente.

Negocio actual

En las entidades empresariales y presupuestadas se llevan a cabo a menudo el proceso Realizar Análisis Financiero; el cual tiene lugar cuando el usuario decide que es necesario realizar un análisis financiero a una entidad económica, para conocer el comportamiento operativo de esta. Para esto, el contador determina un período base y un intervalo de períodos a comparar, además de un estado financiero y el tipo de técnica por el que se realizará el análisis. Luego de esto, se procede a obtener los saldos por cada período de los contenidos y las cuentas pertenecientes al estado financiero seleccionado previamente. Una vez obtenidos los datos necesarios se realizan los cálculos correspondientes a cada técnica de análisis y luego se obtiene un resumen con los resultados del análisis financiero realizado.

2.3. Propuesta del sistema

El sistema deberá ser capaz de realizar análisis financieros en Cedrux mediante técnicas. Para ello deberá permitir que el usuario pueda autenticarse, verificando así sus privilegios, y de esta forma dicho usuario podrá acceder solo a las áreas a las que tenga acceso, así como las funcionalidades y acciones que le están permitidas según su rol. Para la realización de dicho análisis, el usuario previamente autenticado, deberá configurar al menos un estado financiero, el cual puede ser un Estado de Situación o Estado de Resultado. Una vez que sea configurado el estado financiero, se procederá a la realización de su análisis a partir de las técnicas implementadas en el sistema. Luego de haber realizado el análisis, el sistema debe brindarle al usuario un

reporte detallado donde se mostrarán los resultados obtenidos, posibilitando la opción de imprimir.

Técnicas utilizadas para descubrir el negocio

- **Entrevistas:** Inicialmente se realizan entrevistas al cliente que sirven para entender cómo realmente se llevan a cabo las actividades correspondientes al análisis financiero en las entidades cubanas.
- **Storyboard:** Esta técnica fue de gran beneficio, ya que mediante esbozos informales realizados, se pudo llegar a un entendimiento entre el cliente y el equipo de análisis acerca de lo que el proceso Realizar Análisis Financiero debería realizar.
- **Arqueología de documentos:** Se efectuó un estudio de la documentación suministrada por parte del cliente, referente al desarrollo del proceso, que luego sirvió de sustento para confeccionar las descripciones de los procesos de negocio.

2.3. Modelo de negocio del proceso Realizar Análisis Financiero

El modelado de negocio tiene como objetivo describir los procesos existentes u observados, con el propósito de comprenderlos. Los modelos de procesos deben estar expresados de forma que permitan que estos sean analizados, automatizados o mejorados (Layola, 2006). Estos aspectos serán reflejados en este caso mediante el mapa de procesos del negocio, el diagrama de proceso de negocio y su respectiva descripción.

Mapa de procesos del negocio

Se utilizó para organizar en paquetes todos los procesos existentes en el negocio y mostrar cómo se relacionan entre ellos. El mapa de procesos para el componente encargado de la realización del análisis financiero en el subsistema de Contabilidad en Cedrux abarca 2 procesos denominados: Realizar Cierre de Balance y Realizar Análisis Financiero, donde el segundo necesita de los estados financieros que genera el primero. Para más detalle ver la figura 2.

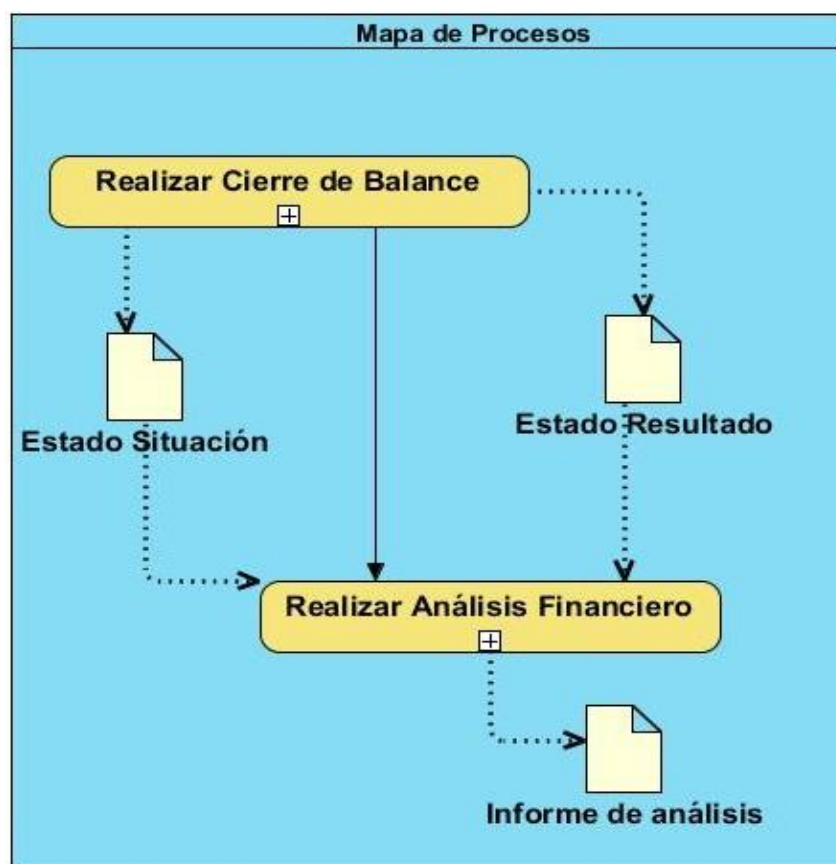


Figura 2 Mapa de proceso de negocio.

Descripción del proceso de negocio

A continuación se muestra la descripción del proceso Realizar Análisis Financiero, donde se podrá explicar detalladamente en qué consiste. La descripción del proceso Realizar Cierre de Balance se puede consultar en el expediente de proyecto de Contabilidad del centro CEIGE.

Proceso Realizar Análisis Financiero

Objetivo	Obtener la información necesaria durante la vida activa de una empresa para observar las debilidades y puntos fuertes que presenta.
Evento(s) que lo genera(n)	N/A.
Pre-condiciones	Debe existir una entidad. Deben existir los estados financieros.
Marco legal	Resolución No. 402-2011 Ministerio de

	<p>Finanzas y Precios.</p> <p>Resolución No. 400-2011 Ministerio de Finanzas y Precios.</p> <p>Resolución No. 433-2012 Ministerio de Finanzas y Precios.</p> <p>Resolución No. 235-05 Ministerio de Finanzas y Precios.</p> <p>Resolución No. 294-2005 Ministerio de Finanzas y Precios.</p> <p>Resolución No. 486-2013 Ministerio de Finanzas y Precios.</p> <p>Resolución No. 426-2012 Ministerio de Finanzas y Precios.</p>
Clientes externos	Órganos de Finanzas.
Entradas	Estado de situación, Estado de resultados.
Flujo de eventos	
Flujo básico Análisis Financiero	
1.	Determinar período o ejercicio: El contador, para realizar cada técnica y realizar el análisis, necesita primeramente determinar si lo realizará por ejercicio o período.
2.	Determinar ejercicio base: Seguidamente el contador determinará un ejercicio base.
3.	Determinar intervalo de ejercicios: el contador elegirá ejercicios a comparar.
4.	Determinar período base: Seguidamente el contador optará por un período base.
5.	Determinar intervalo de períodos: el contador determina un intervalo de períodos que le servirán para la comparación.

6.	Determinar tipo de estado financiero a analizar: El contador selecciona el estado financiero necesario para aplicar el análisis, el cual puede ser: Estado Situación o Estado Resultado.
7.	Determinar el tipo de técnica: A continuación se determina por cuál, de las técnicas del análisis, se realizará el análisis financiero en la entidad.
8.	Entregar informe del análisis: Luego de haber realizado el análisis se hace entrega de un informe con los resultados del mismo al Director de la entidad.
Post-condiciones	
1.	Se obtiene un informe con los datos generados por el análisis financiero.
Salidas	
1.	Reporte con los resultados del análisis.
Flujos paralelos	
1.	Determinar intervalo de ejercicios: el contador elegirá para períodos a comparar pertenecientes a sus respectivos ejercicios.
2.	Determinar período base: Seguidamente el contador seleccionará un período base.
3.	Determinar intervalo de períodos: el contador determina un intervalo de períodos que le servirán para la comparación.
Salidas	
N/A	
Flujos alternos	
Flujo alternativo 1.a Realizar Cambios en pesos y porcentajes	
1. Se realiza esta técnica calculando primeramente el cambio en peso, dividiendo el valor del saldo del año base entre un valor del año de comparación.	
2. Luego se divide el valor de esta división por el valor del año base.	
Flujo alternativo 2.a Realizar Porcentajes de tendencias	
1. Según un año base, se le da un valor del 100% a cada partida de los estados financieros.	

<p>2. Se expresa cada partida de los estados financieros de los años siguientes como un % del valor del año base.</p>
<p>3. Se divide cada partida, en los años siguientes del año base por el valor del año base.</p>
<p>Flujo alterno 3.a Realizar Porcentajes componentes.</p>
<p>1. El contador primeramente, para la realización de esta técnica, toma para un período, una partida que será la partida base, la cual servirá de comparación para ver si las demás partidas se van convirtiendo menos significativas, es decir cuáles van en descenso. A esta partida inicial le da un valor del 100 %.</p> <p>2. El contador, a partir de esta partida inicial, toma otra partida a la cual calculará el porcentaje que esta otra representa, y así vera los cambios significativos.</p>
<p>Flujo alterno 4.a Realizar Cálculo del equilibrio financiero.</p>
<p>1. Para la realización de esta técnica el contador compara los Activos Circulantes con los Pasivos Circulantes para ver si existe solvencia y liquidez, conociéndose que una empresa posee liquidez si su activo circulante es mayor que su pasivo circulante y es solvente si su activo real (activos fijos y activos circulantes) es mayor que las deudas de la empresa, a corto, mediano y largo plazo, es decir, los financiamientos ajenos.</p>
<p>2. El contador necesita además analizar si existe estabilidad comprobando las deudas de la empresa, a corto, mediano y largo plazo, es decir los financiamientos ajenos y el capital de la empresa también conocidos como financiamientos propios. Si el porcentaje del financiamiento ajeno obtenido es mayor que el valor límite superior especificado por el usuario, y el porcentaje del financiamiento propio es menor que el valor límite inferior especificado también por el usuario, entonces se considera inestable, pero en caso de encontrarse los porcentajes dentro del rango definido, entonces se considera estable.</p>
<p>Flujo alterno 5.a Realizar Variación de capital de trabajo</p>
<p>1. Para realizar este análisis primeramente el contador debe agrupar las diferentes partidas en Activo circulante y Pasivo circulante, partiendo de que el activo circulante es el grupo de cuentas que representan bienes o derechos que convertirán en efectivo en un corto período de tiempo y los pasivos circulantes son el grupo de cuentas que reflejan deudas y obligaciones en un corto período de tiempo.</p>

<p>2. Luego contador comprueba que exista liquidez o sea, que los Activos Circulantes sean mayores que los Pasivos Circulantes. Se calcula los aumentos y disminución de los activos y los pasivos.</p>
<p>3. El contador debe seguir reglas para la determinación de la variación de capital de trabajo, las cuales son:</p> <ul style="list-style-type: none"> • Aumentos de Capital de Trabajo (Aumentos de Activos Circulantes y Disminuciones de Pasivos Circulantes). • Disminución de Capital de Trabajo (Disminución de Activos Circulantes y Aumentos de Pasivos Circulantes). • Aumento Neto de Capital de Trabajo (Aumentos de Capital de Trabajo mayores que las Disminuciones de Capital de Trabajo). • Disminución Neta de Capital de Trabajo (Disminuciones de Capital de Trabajo mayores que Aumentos de Capital de Trabajo).
<p>Flujo alterno 6.a Realizar Graficar los estados financieros</p>
<p>1. Con esta técnica el contador tiene la posibilidad de mostrar los datos contables del análisis en diagramas, para así obtener una visión más clara de los problemas.</p>
<p>2. Se debe conocer que cuando se analiza el gráfico de un solo balance se le llama análisis estático y cuando se analiza el de varios balances, se le llama análisis dinámico.</p>
<p>3. El contador, a partir de los porcentos calculados en la técnica Tamaño común o Porcentaje componente, obtiene el gráfico de balance, donde se realizará un rectángulo, el cual dividirá en dos partes iguales, considerando siempre que la altura de este es de 100%.</p>
<p>Salidas</p>
<p>N/A</p>
<p>Asuntos pendientes</p>
<p>N/A</p>

Tabla 4 Descripción del proceso Realizar Análisis Financiero.

Diagrama del proceso de negocio

A continuación se muestra el proceso Realizar Análisis Financiero (ver figura 3), que se describe posteriormente y el proceso Cierre de Balance se puede observar en el expediente de proyecto.

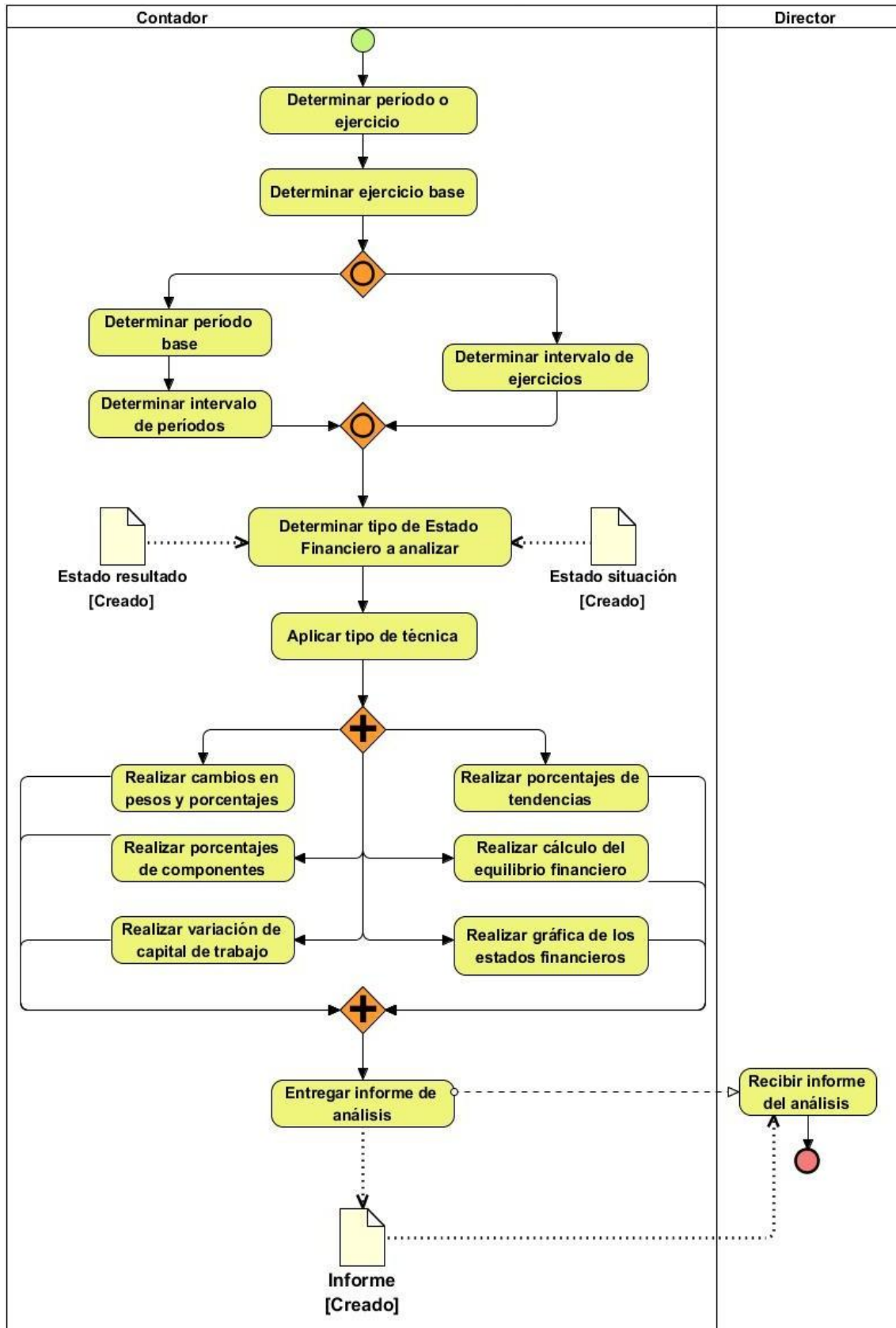


Figura 3 Diagrama del proceso de negocio: Realizar Análisis Financiero.

Modelo Conceptual

El modelo conceptual o modelo de dominio como también se le conoce (Ver Figura 4), permite de manera visual mostrar al usuario los principales conceptos que se manejan en el dominio del problema. Es una representación de las clases conceptuales del mundo real, no de componentes de software (Unidad de Compatibilización Integración y Desarrollo, 2009).

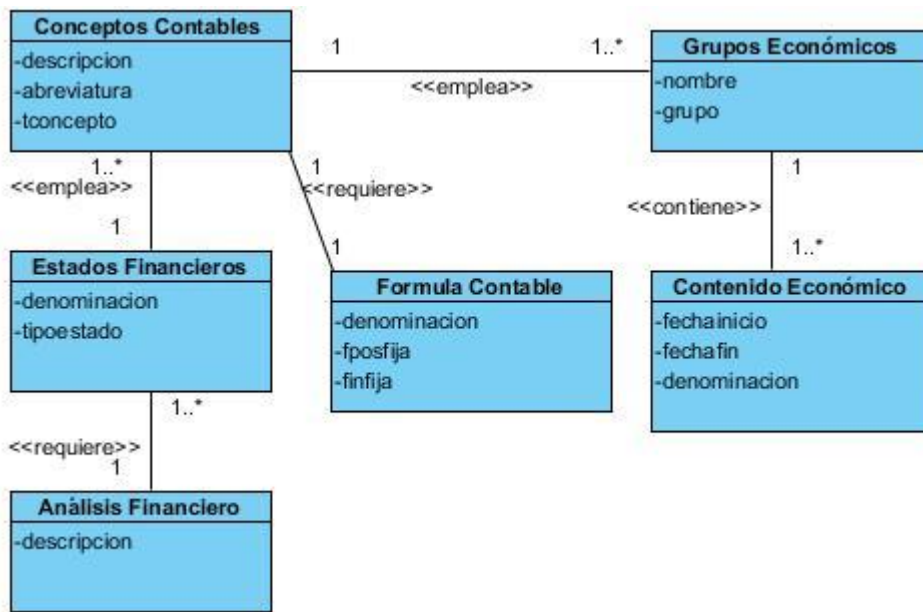


Figura 4 Modelo Conceptual.

2.4. Ingeniería de requisitos asociada al proceso Realizar Análisis Financiero

El propósito de la definición de requisitos es especificar las condiciones o capacidades que el sistema debe cumplir y las restricciones bajo las cuales debe operar, según lo indicado por (Colectivo de autores: UCID, 2009).

Requisitos funcionales

Para el levantamiento de requisitos del componente Análisis Financiero se utilizaron las técnicas de captura de los mismos definidos en el capítulo anterior. Primeramente se desarrollaron varios encuentros con el cliente, y haciendo uso de la tormenta de ideas se obtuvo una visión general de las características que debía tener el componente. Además en numerosas entrevistas con el cliente y con un funcional con vasto conocimiento en el tema, se pudieron detallar cada una de las funcionalidades que el sistema debía cumplir. Por otra parte se revisaron algunos documentos y bibliografías de prestigio relacionadas con los análisis financieros que se realizan en

las entidades, dentro de las cuales se encontraron métodos y técnicas que contribuyen a la correcta realización del análisis para una mejor visión de la conducta operativa de una empresa. Se analizaron sistemas existentes, inspeccionando las interfaces de usuario además de la información que se manejaba en ellas y cómo se hacía. También se estudiaron salidas de dichos sistemas haciendo énfasis en la forma en que se mostraban los reportes y listados, con el propósito de que surgieran ideas que apoyaran la implementación.

A partir del proceso de negocio anteriormente analizado y con el apoyo que brindaron las diferentes técnicas de captura de requisitos empleadas, se llegó a la conclusión de que existirán dos requisitos funcionales donde el primero contendrá 6 flujos alternativos que corresponden a cada una de las técnicas del análisis, quedando de la siguiente manera:

- ✓ **Configurar análisis financiero.**
- ✓ **Emitir reporte de análisis financiero.**

Para configurar un análisis financiero primeramente en la interfaz principal, el usuario tendría la opción de seleccionar si realizará dicho análisis por período o ejercicio, para con esto seleccionar en dependencia de las opciones un ejercicio y un período base y luego un intervalo de ejercicios o períodos a comparar. Seleccionará posteriormente el estado financiero al que le realizará el análisis y por último la técnica por la que se realiza el análisis financiero. Una vez configurado el análisis se procede a emitir un reporte con los resultados del mismo, el cual tendrá un formato determinado para cada técnica seleccionada. Para ello el usuario deberá seleccionar la opción de Imprimir Reporte mostrándose una interfaz con el formato especificado.

Patrones aplicados

- **Completar una única meta:** cuando se identificaban los requisitos, se comprobó que los mismos estuvieran dirigidos hacia una completa y bien precisada meta.
- **El nombre revela la intención:** los requisitos fueron nombrados teniendo en cuenta que su nombre relevaba su intención. Para ello se decidió que la primera palabra fuese un verbo y la segunda un sustantivo relacionado con la operación asociada al requisito. Esto permitió que al leer su nombre se determinara el propósito del mismo.

- **Preciso y legible:** la especificación de cada requisito se realizó de forma clara y específica, comunicando la información suficiente para su adecuada comprensión.

Especificación de los requisitos funcionales

Como indica (Moreno, 2009), “las especificaciones de requisitos son indicaciones textuales de los mismos. Tienen el formato de un documento relacionado con los requisitos y explica cómo se lleva a cabo el requisito”. A continuación se muestra la descripción del requisito funcional Configurar Análisis Financiero, las especificaciones restantes se encuentran en el expediente de proyecto del centro CEIGE, además de los restantes artefactos entregados del componente.

Especificación del requisito Configurar Análisis Financiero.

Precondiciones	El usuario se debe haber autenticado en el sistema. Deben existir estados financieros definidos en el sistema.
Flujo de eventos	
Flujo básico	
1 Se seleccionan los datos necesarios para la realización del análisis financiero. Se muestra: <ul style="list-style-type: none"> ✓ Período o ejercicio, donde en caso de que la elección fuese ejercicio se selecciona un ejercicio base, en caso de elegir período, se selecciona para un ejercicio base y para este un período base. ✓ Períodos o ejercicios a comparar, donde sí se selecciona ejercicio, se elige además un intervalo de ejercicios que para comparar y si selecciona período se elige un intervalo de períodos correspondientes a un ejercicio. En cualquiera de los dos casos el sistema valida la cantidad de períodos o ejercicios a comparar (ver validación 1). ✓ Tipo de estados financiero: Estado Resultado o Estado Situación. ✓ Técnicas a escoger para realizar dicho análisis. 	
2 El sistema valida que todos los datos hayan sido seleccionados (ver validación 2).	
3 Si los datos son correctos, entonces el sistema los registra.	
4 El sistema confirma el registro de los datos.	
5 Concluye el requisito.	
Post-condiciones	
1 Se registró en el sistema una nueva información para la realización del análisis	

financiero según la técnica seleccionada.
Flujos alternativos
Flujo alternativo 1.a La técnica seleccionada es Cambios en pesos y porcentajes.
1. El sistema a partir de los datos seleccionados realiza el análisis.
2. Volver al paso 1 del flujo básico.
3. Concluye el requisito.
Post-condiciones
1. Se realiza el cambio en pesos y porcentajes.
Flujo alternativo 1.b La técnica seleccionada es Porcentajes de tendencias.
1. El sistema a partir de los datos seleccionados realiza el análisis.
2. Volver al paso 1 del flujo básico.
3. Concluye el requisito.
Post-condiciones
1. Se realiza la técnica Porcentajes de tendencias.
Flujo alternativo 1.c La técnica seleccionada es Porcentajes de componentes.
1. El sistema a partir de los datos seleccionados, realiza el análisis.
2. Volver al paso 1 del flujo básico.
3. Concluye el requisito.
Post-condiciones
1. Se realizó la técnica Porcentajes de componentes.
Flujo alternativo 1.d La técnica seleccionada es Calculo del equilibrio financiero.
1. Se levanta una nueva ventana donde se seleccionan e introducen otros datos necesarios para esta técnica.
2. Se seleccionan los activos y pasivos circulantes, activos reales, financiamientos ajenos y propios (ver validación 2).
3. Se introducen además los valores de los financiamientos ajenos y propios para ver si existe estabilidad
4. El sistema verifica la entrada de los datos (ver validación 3).
5. El sistema realiza el análisis.
6. Volver al paso 1 del flujo básico.
7. Concluye el requisito.

Post-condiciones
1. Se realizó la técnica Cálculo de equilibrio financiero.
Flujo alternativo 1.f La técnica seleccionada es Variación de capital de trabajo.
1. Se levanta una ventana para seleccionar los activos, pasivos circulantes y año necesarios (ver validación 1).
2. El sistema a partir de los datos seleccionados realiza el análisis.
3. Volver al paso 1 del flujo básico.
4. Concluye el requisito.
Post-condiciones
1. Se realizó la técnica Variación de capital de trabajo.
Flujo alternativo 1.g La técnica seleccionada es Graficar los estados financieros.
1. Una vez que se selecciona esta técnica el sistema habilita los botones para elegir tipo de gráfica, valor a representar en la gráfica y el año o años por el cual o los cuales se graficará.
2. El sistema a partir de los datos seleccionados anteriormente realiza el análisis.
3. Concluye el requisito.
Post-condiciones
1. Se realizó la técnica Graficar los estados financieros.
Flujo alternativo 2.a Información incompleta
1 El sistema señala los datos erróneos y permite corregirlos.
2 El usuario corrige los datos.
3 Volver al paso 1 del flujo básico.
Post-condiciones
N/A
Flujo alternativo *.a El usuario cancela la acción
1 Concluye el requisito.
Post-condiciones
1 No se registran los datos.

Validaciones	
6	En caso de que la cantidad de ejercicio fuera mayor de 3, el sistema muestra un mensaje indicándole al usuario que la cantidad escogida no es la aceptada.
7	El sistema verifica si fueron seleccionados todos los datos. En caso de no haberse seleccionado algunos de estos parámetros, el sistema responde con un mensaje de alerta informando que falta un parámetro por seleccionar, se pasa al paso 1 del flujo básico.
8	Se validan los datos entrados, mostrando un mensaje en caso de que se entren datos incorrectos o se dejen vacíos los campos.
Relaciones	Requisitos Incluidos
	Extensiones
Conceptos	<p>Activo circulante: Son aquellas cuentas dentro del balance general que son rápidamente convertibles a efectivo. Entre las cuentas más comunes están: efectivo, inversiones temporales, inventario, clientes, deudores diversos, bancos, etc.</p> <p>Pasivo circulante: Cuentas del Balance que representan el exigible a corto Plazo.</p>
Requisitos especiales	N/A
Asuntos pendientes	N/A.

Tabla 5 Especificación del requisito Configurar Análisis Financiero.

Prototipo de interfaz de usuario Configurar Análisis Financiero

A continuación se muestra el prototipo de interfaz de usuario Configurar Análisis Financiero (Ver figura 5), algunos de los prototipos correspondientes a dicho requisito pueden consultarse en el Anexo 1.

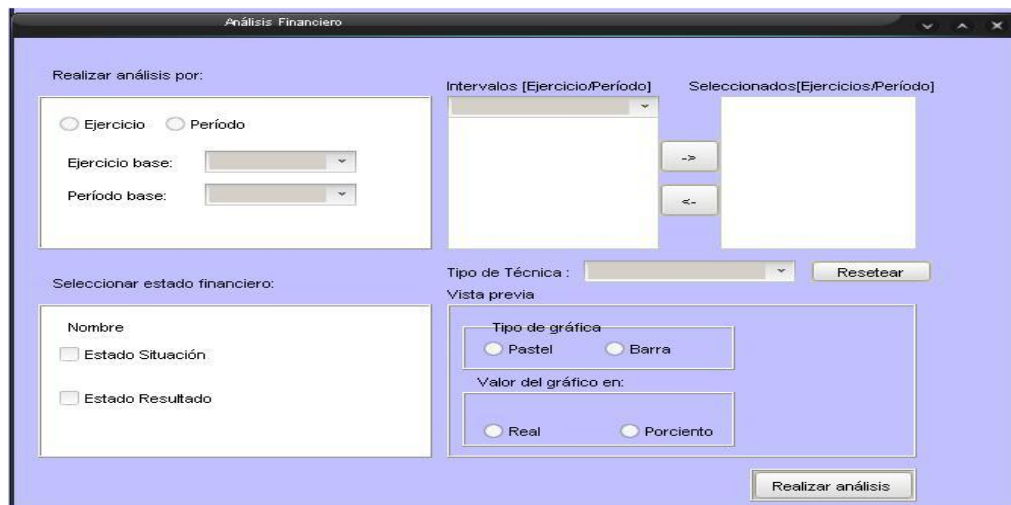


Figura 5 Prototipo de IU Configurar Análisis Financiero.

2.5. Diseño de la solución

Modelo de datos

El modelo de datos propuesto en la solución emplea un total de 13 tablas, que son las necesarias para la realización del requisito Configurar análisis financiero, a continuación se muestra dicho modelo, resaltando la tabla nom_estadof, la cual será descrita posteriormente (Ver Figura 6).

Los estados financieros, corazón del Análisis financiero, presentarán una relación de muchos a muchos con los conceptos contables, para de esta forma hacer posible que un estado financiero contenga varios conceptos contables y a su vez un concepto contable pueda estar contenido dentro de varios estados financieros. Los contenidos económicos forman parte de la fórmula de los conceptos contables, y también tienen el valor de la suma de sus cuentas, que es necesario para poder realizar los cálculos correspondientes en cada una de las técnicas de análisis, es por ello que los estados financieros y los conceptos contables tendrán una relación indirecta con la tabla formula_nodo, mediante la tabla dat_fnodoestadofcc, en donde se guardarán los identificadores de la fórmula correspondiente a los contenidos de cada concepto, así como su respectivo plan.

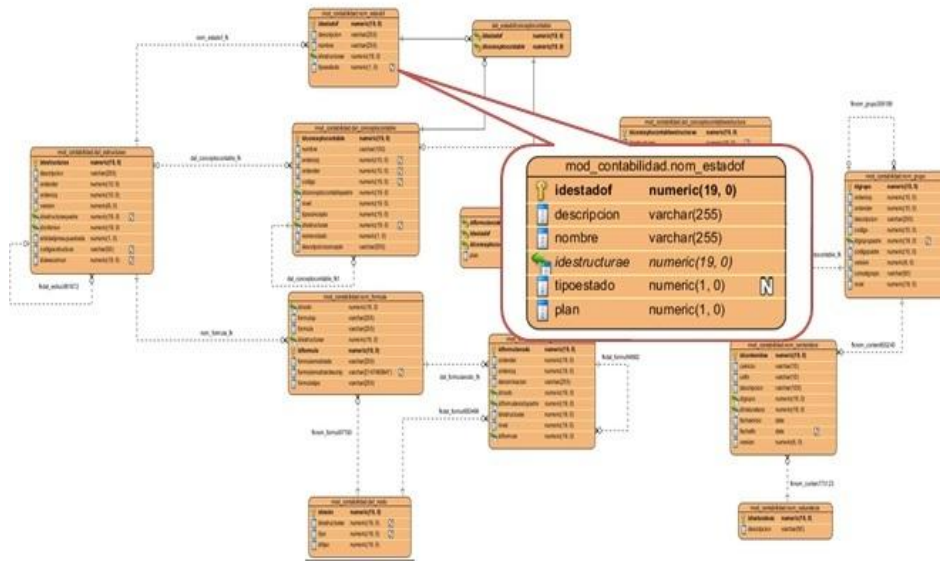


Figura 6 Modelo de Datos.

Descripción de las tablas utilizadas

Se muestra la descripción de una de las tablas utilizadas (Ver Tabla 2), que es la perteneciente a los estados financieros. Esta tabla se creó debido a la importancia de almacenar la información de estos para su posterior análisis. La ampliación del modelo de datos y las restantes descripciones, se pueden observar en el diccionario de datos que se encuentra en el expediente de proyecto, en conjunto con los demás artefactos generados.

Nombre: mod_contabilidad.nom_estadof		
Atributo	Tipo	Descripción
idestadof	numeric	Llave primaria de la tabla.
descripcion	varchar	Descripción del estado financiero.
nombre	varchar	Nombre abreviado del estado financiero, que identifica a este.
idestructurae	numeric	Llave foránea, identificador de la entidad la cual pertenece el estado financiero.

tipoestado	numeric	Identificador que muestra que tipo de estado es el estado (estado situación o estado de resultados).
Tablas Relacionadas	Descripción de la relación	
mod_contabilidad.dat_conceptocontable	Mucho a muchos.	
Descripción de la tabla		
Almacena los datos correspondientes a los estados financieros.		

Tabla 6 Diccionario de datos tabla mod_contabilidad.nom_estadof.

Diagrama de clases del diseño

Como plantea (Arizaca, 2009) el Diagrama de Clases es uno de los diagramas principales para el análisis y diseño de un software, ya que sirve para visualizar las distintas relaciones estructurales y de herencia. Incluye además definiciones para atributos y operaciones. En el presente trabajo se muestra el diagrama de clases correspondiente al proceso Realizar Análisis financiero del componente a implementar, (Ver Figura 7), también se puede apreciar el diagrama de clases para el proceso Gestionar Estados Financieros necesario para realizar el análisis, en el expediente de proyecto del centro CEIGE.



Figura 8 Aplicación del patrón Experto.

Creador: Este patrón es adaptable a las clases del paquete Domain, quienes son las encargadas de crear los objetos de tipo Doctrine_Query, para permitir el acceso a la información almacenada a nivel de datos.

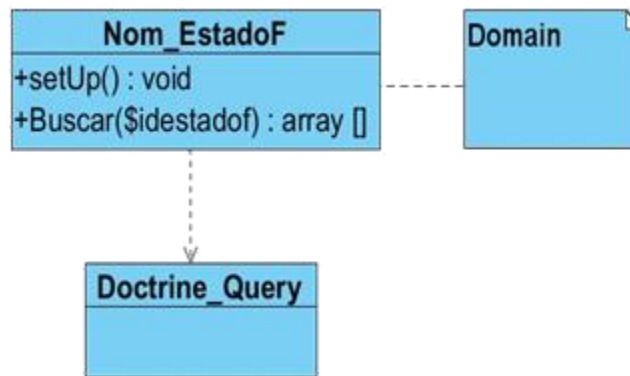


Figura 9 Aplicación del patrón Creador.

Controlador: La clase controladora mencionada anteriormente, es un ejemplo de la aplicación de este patrón, la cual se encargará de manejar los eventos dentro del componente.

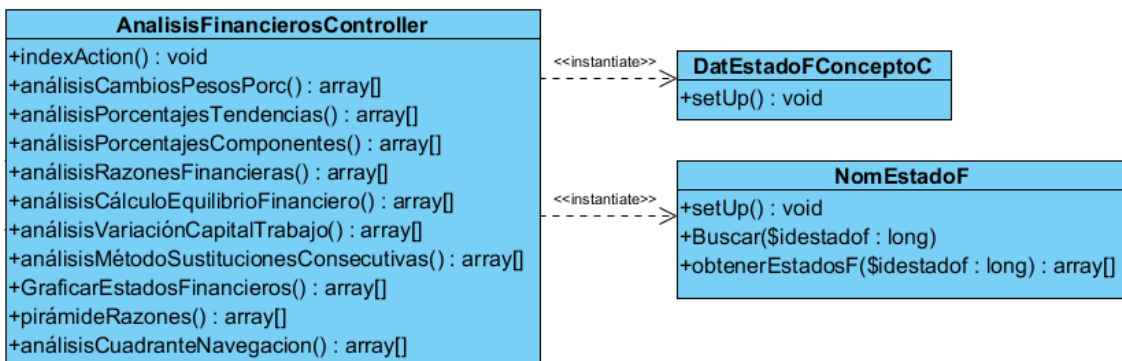


Figura 10 Aplicación del patrón Controlador.

Diagramas de secuencias

Los diagramas de interacción modelan los aspectos dinámicos de un sistema y muestran un perspectiva general del flujo de control dentro del sistema o proceso de negocio (Ávila, 2009)

Por tanto se realizaron diagramas de secuencia, que son un tipo de diagrama de interacción que modela la secuencia lógica de una acción ejecutada por un usuario y el orden en que suceden los mensajes. A continuación se muestra el diagrama de secuencia para realizar el análisis financiero por la técnica Cambio en pesos y porcentajes (Ver Figura 11), los restantes diagramas de secuencias pueden ser consultados en el expediente de proyecto del centro.

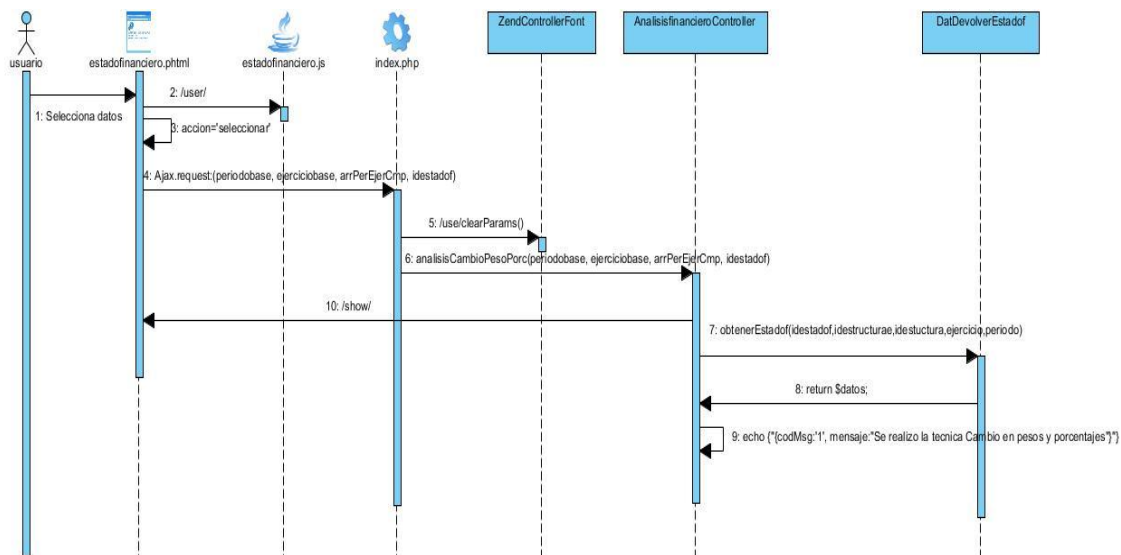


Figura 11 Diagrama de secuencia para Cambio en pesos y porcentajes.

2.6. Implementación de la solución

Diagrama de Componentes

Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes, además de ayudar a entender mejor el modelo de implementación. A continuación se explica la forma de interacción del componente Análisis Financiero con los componentes Estructura y Composición, Seguridad, Configuración y Estados Financieros, para un mayor entendimiento del diagrama (Ver Figura 11).

Componente Análisis Financiero: Es el encargado de realizar el análisis financiero a los estados financieros obtenidos del componente Estado Financiero. También recibe

la estructura común del componente Estructura y Composición, además de consumir los servicios de autenticación del componente Seguridad. Por otra parte recibe la fecha, el período y el ejercicio contable del componente de Configuración General.

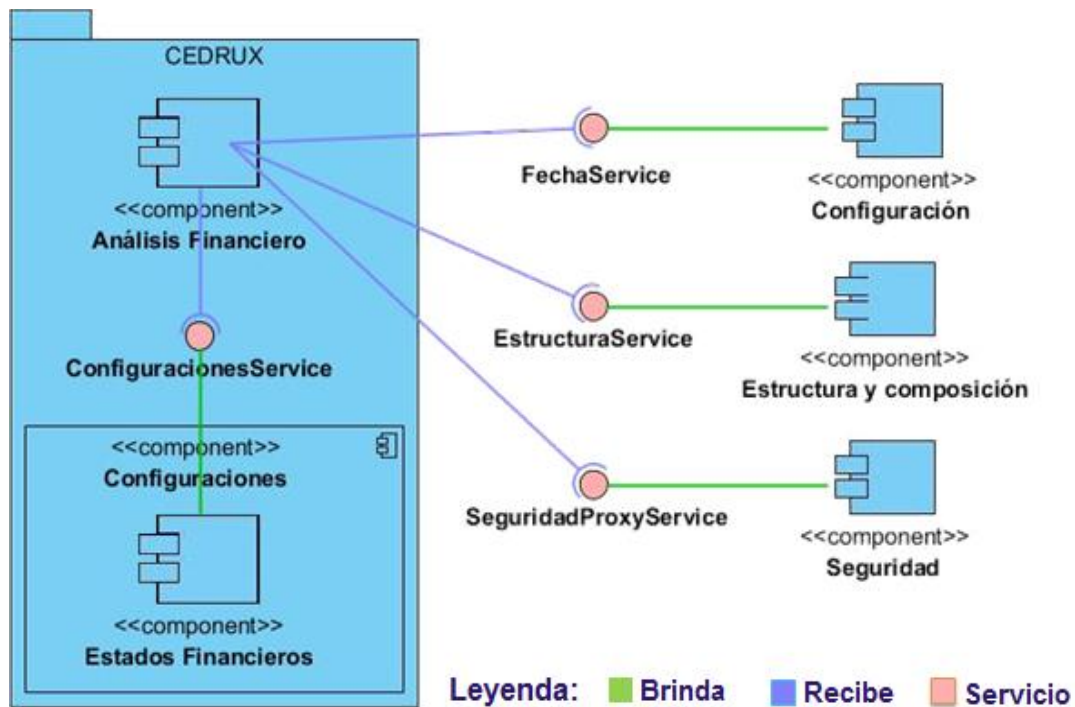


Figura 12 Diagrama de componente.

Prototipo de interfaz de usuario funcional

Los prototipos de interfaz de usuario brindan la posibilidad de realizar la validación de los requisitos funcionales en cuanto a las funcionalidades o condiciones de ejecución que deben cumplirse.

A continuación se muestra el prototipo de interfaz de usuario funcional del requisito Configurar Análisis Financiero que se obtuvo como resultado del diseño del sistema (Ver Figura 9). Algunos de los prototipos de interfaz de usuario funcional, se pueden consultar en el Anexo 2.

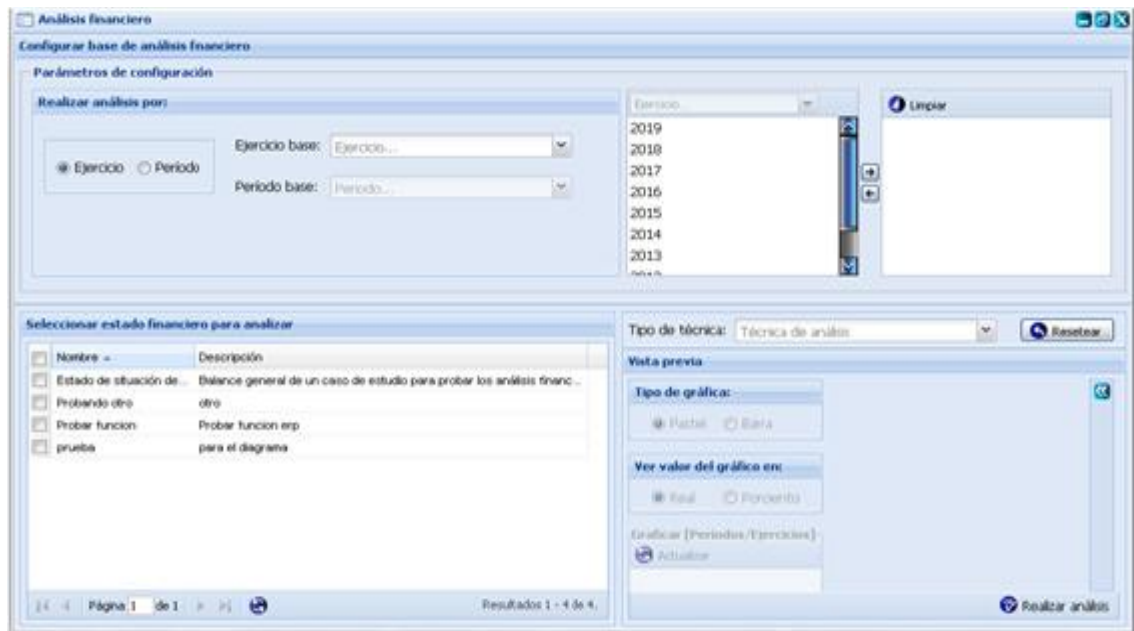


Figura 13 Prototipo de IU funcional Configurar Análisis Financiero.

2.7. Conclusiones parciales

Entre los principales resultados alcanzados durante el desarrollo del capítulo se resaltan los siguientes:

- La propuesta de solución, en conjunto con el mapa y el diagrama de procesos del negocio, posibilitaron comprender como funciona en las entidades, el proceso que es necesario informatizar.
- Se llega a un convenio sobre el proceso que debe ser informatizado, lo que representa para los desarrolladores la base fundamental para identificar el sistema que realmente necesita el cliente.
- Mediante la realización del levantamiento de requisitos y con las estrategias de obtención de requisitos puestas en práctica, se consiguió identificar las funcionalidades que el componente Análisis Financiero debe brindar, siendo fieles a las necesidades del cliente.
- El diagrama de clases del diseño permitió conocer la estructura y las relaciones entre las clases que se manejan en el componente, posibilitando a su vez, el mantenimiento y entendimiento de estas, mediante los patrones de diseño definidos en la estructura arquitectónica de la solución.
- A partir de los artefactos del análisis y el diseño, se logra fundamentar las bases para la implementación del componente.

CAPÍTULO 3. VALIDACIÓN DE LA SOLUCIÓN PROPUESTA

3.1. Introducción

En el presente capítulo se realiza la validación de la propuesta de solución plasmada en el capítulo anterior. Se validan los requisitos y el diseño, permitiendo mostrar que dichos requisitos definen el sistema que el usuario desea y que el diseño realizado cumple con los patrones utilizados, por lo que se aplican técnicas y métricas para mostrar la calidad del componente.

3.2. Validación del modelo de negocio e ingeniería de requisitos

Validación del negocio

Luego de identificado y descrito el proceso de negocio, se procedió a la validación del mismo, con el objetivo de asegurarse de que se realizó correctamente la identificación de este y que el equipo entiende con claridad el negocio que se desea informatizar, para esto fue necesario reunirse nuevamente con el cliente. En estos encuentros se discutieron los modelos realizados, donde el cliente verificó que la modelación del proceso se correspondía al desarrollo de dicha actividad en las entidades cubanas. Las descripciones de los modelos aprobados por el cliente pasaron a ser revisadas por el especialista de calidad de la línea Contabilidad, en donde se verificó el cumplimiento de los estándares establecidos para la descripción, logrando que esta sea entendible para los implicados. Por tanto el proceso quedó validado y aprobado por el cliente obteniéndose una carta de aceptación del mismo.

En la tabla siguiente (Ver Tabla 3), se muestra el proceso que fue validado teniendo en cuenta algunos parámetros que se especifican en una herramienta elaborada por el Centro de Soluciones de Gestión, la cual ofrece varios criterios para validar procesos de negocio.

Solicitado por	Msc. Joisel Pérez Pérez	Proceso de negocio
Fecha de solicitud	04/02/2013	
Criterios para validar	Realizar Análisis Financiero	

¿El proveedor del proceso es un proveedor válido?	Sí
¿El proceso está unívocamente identificado?	Sí
¿El proceso de negocio está completo?	Sí
¿El proceso de negocio es consistente?	Sí
¿El resultado de la evaluación de impacto es positivo?	Sí
¿El proceso está correcto según los proveedores de información?	Sí
Decisión	APROBADO

Tabla 7 Validación del Diseño.

Validación de los requisitos

Una vez identificados los requisitos, se procede a validar los mismos, pues un levantamiento de requisitos con errores que no se detecten a tiempo provoca la pérdida de tiempo y conduce a resultados inesperados. Esta validación se realizó mediante las dos técnicas seleccionadas en el Capítulo 1. La Revisión técnica formal a los documentos y artefactos por parte del cliente y el equipo de desarrollo fue una de las técnicas utilizadas en la validación, con lo que se pudo comprobar que la información brindada por el cliente permitió realizar una correcta especificación de los requisitos que el componente debe presentar. La técnica Prototipo de interfaz de usuario fue muy utilizada, debido a que en los prototipos se engloban las funcionalidades del sistema y el cliente pudo apreciar una aproximación al producto. Finalmente se produjo la aceptación de los requisitos por parte del cliente, el Msc. Joisel Pérez Pérez, el analista del proyecto Ing. Yordano Yuniór Osorio Cintra y la jefa de la línea Ing. Yanay Hernández Sosa.

3.3. Validación del diseño propuesto

Con el objetivo de validar que el diseño de clases propuesto está correcto se emplean las métricas Tamaño operacional de clase y Relaciones entre clases seleccionadas en el Capítulo 1.

Tamaño operacional de clase

Esta métrica evalúa los siguientes atributos de calidad:

Atributo de Calidad	Modo en que lo afecta
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 en la complejidad de implementación de la clase.
Reutilización	Un aumento del TOC implica una disminución del grado de reutilización de la clase.

Tabla 8 Atributos de calidad que evalúa la métrica TOC.

Luego de aplicar esta métrica para un total de 13 clases, los valores umbrales obtenidos que representan la incidencia de los resultados de la evaluación en cada uno de los atributos son los siguientes:

Atributo	Categoría	Criterio	Umbral
Responsabilidad	Baja	(\leq Prom)	69.23
	Media	(Entre Prom y 2^* Prom)	23.08
	Alta	($> 2^*$ Prom)	7.69
Complejidad de implementación	Baja	(\leq Prom)	69.23
	Media	(Entre Prom y 2^* Prom)	23.08
	Alta	($> 2^*$ Prom)	7.69
Reutilización	Baja	($> 2^*$ Prom)	7.69
	Media	(Entre Prom y 2^* Prom)	23.08
	Alta	(\leq Prom)	69.23

Tabla 9 Resultados al aplicar la métrica TOC.

Una vez aplicado el instrumento de medición de la métrica TOC y analizados los resultados obtenidos, se concluye que el diseño propuesto está entre los límites aceptables de calidad; teniendo en cuenta que aproximadamente el 69 % de las clases tienen la responsabilidad y complejidad de implementación baja, mientras que un aproximado del 69 % poseen una reutilización alta. Por tanto de esta manera se demuestra que los atributos de calidad evaluados se encuentran en un nivel satisfactorio, ya que se puede observar cómo se fomenta la Reutilización, mecanismo clave en el proceso de desarrollo de software, y cómo están reducidas en menor grado la Responsabilidad y la Complejidad de implementación. Esto posibilita que las clases sean fáciles de implementar, puedan ser reutilizadas en otros diseños y que no sean

complejas, al contener solo las funciones que responden a sus responsabilidades. En la siguiente figura se puede observar con más detalle lo anteriormente explicado.

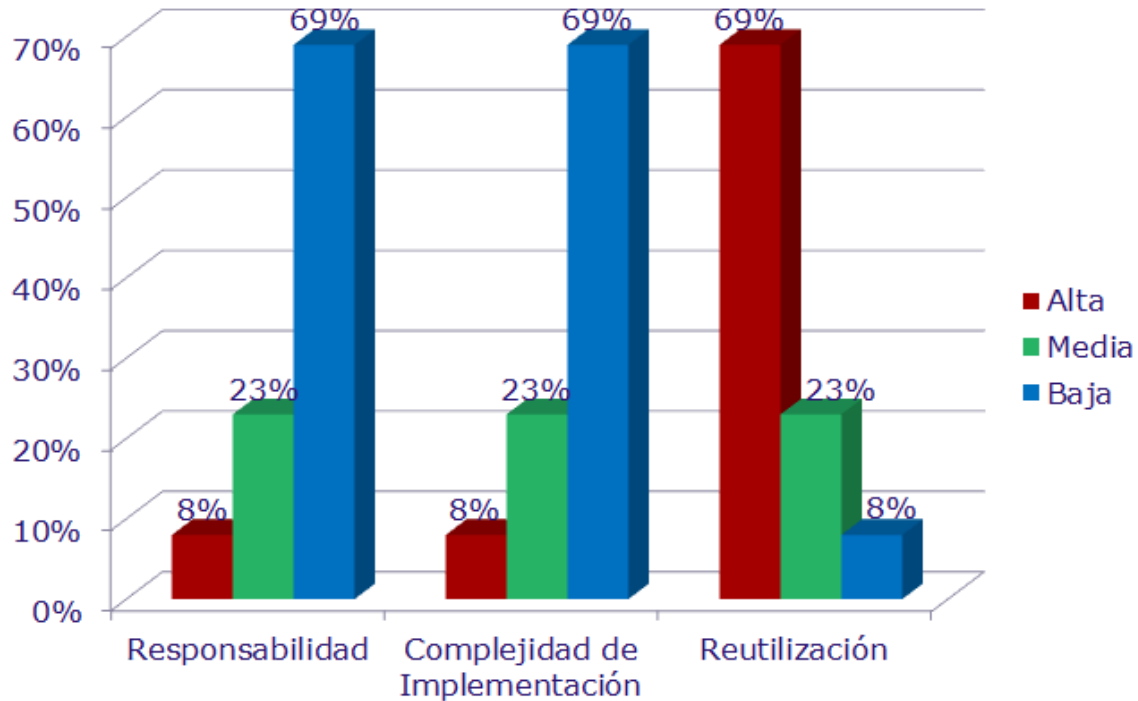


Figura 14 Gráfica de la métrica TOC.

Relaciones entre clases

Esta métrica evalúa los siguientes atributos de calidad:

Atributo de Calidad	Modo en que lo afecta
Acoplamiento	Un aumento del RC implica un aumento del acoplamiento de la clase.
Complejidad de mantenimiento	Un aumento del RC implica un aumento en la complejidad del mantenimiento de la clase.
Reutilización	Un aumento del RC implica una disminución del 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 10 Atributos de calidad que evalúa la métrica RC.

Luego de aplicar esta métrica para un total de 13 clases, los valores umbrales obtenidos que representan la incidencia de los resultados de la evaluación en cada uno de los atributos son los siguientes:

Atributo	Categoría	Criterio	Umbral
Acoplamiento	Ninguno	0	23.08
	Bajo	1	7.69
	Medio	2	53.85
	Alto	(>2)	23.08
Complejidad de mantenimiento	Baja	(<=Prom)	84.62
	Media	(Entre Prom y 2*Prom)	15.38
	Alta	(>2*Prom)	0
Reutilización	Baja	(>2*Prom)	0
	Media	(Entre Prom y 2*Prom)	15.38
	Alta	(<=Prom)	84.62
Cantidad de pruebas	Baja	(<=Prom)	84.62
	Media	(Entre Prom y 2*Prom)	15.38
	Alta	(>2*Prom)	0

Tabla 11 Resultados al aplicar la métrica RC.

Una vez aplicado el instrumento de medición de la métrica RC y analizados los resultados obtenidos, se concluye que el diseño propuesto tiene una calidad aceptable. Los atributos de calidad de encuentran en un nivel satisfactorio; teniendo en cuenta que aproximadamente el 54 % de las clases poseen un nivel medio de acoplamiento, un aproximado del 85 % de baja complejidad de mantenimiento, baja cantidad de pruebas y alta reutilización. Por tanto favorece al mantenimiento de las clases, la cantidad de pruebas a realizar sea mínima, existan pocas dependencias entre ellas y puedan ser reutilizadas en otros diseños. En la siguiente figura se puede observar con más detalle lo anteriormente explicado.

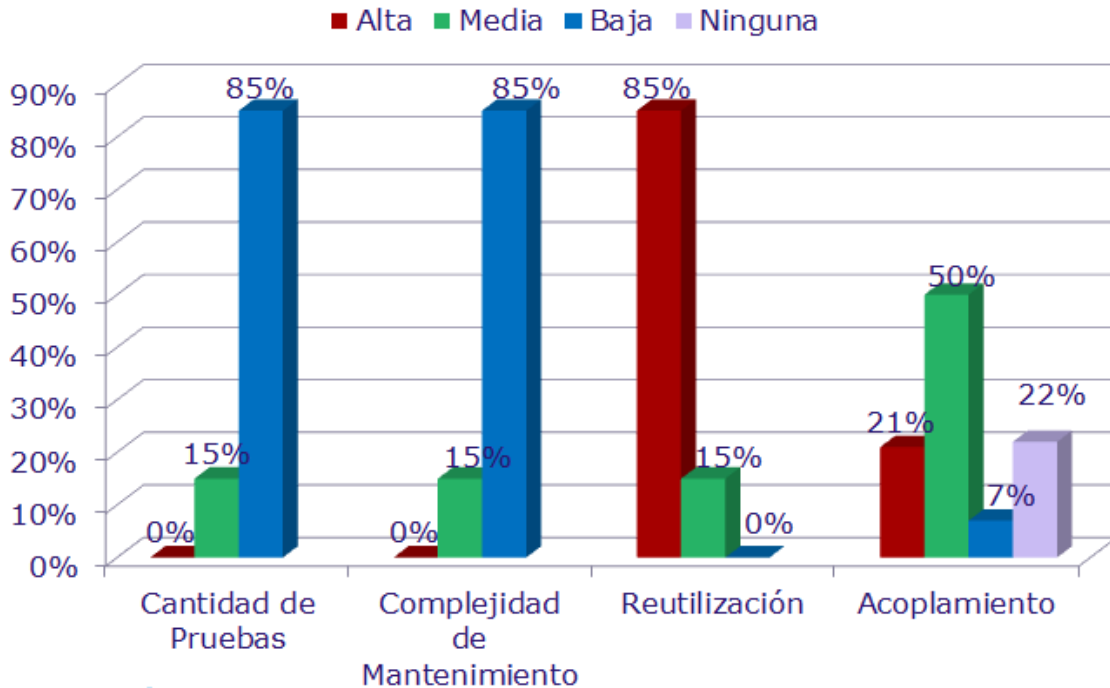


Figura 15 Gráfica de la métrica RC.

3.4. Validación de la aplicación

El comportamiento humano no es perfecto y debido a que el desarrollo de un software está sujeto a una gran cantidad de actividades protagonizadas por el hombre, existe la posibilidad de que aparezcan errores en los sistemas. Por tanto surge la necesidad de realizar pruebas para detectar y corregir posibles errores, que constituyen una etapa imprescindible durante el proceso de desarrollo de un software.

Pruebas

Es importante considerar que las pruebas de software no garantizan que un sistema esté libre de errores, sino que se detecten la mayor cantidad de defectos posibles para su debida corrección, según lo indicado por (Colectivo de autores: UCID, 2009).

Satisfacción de las necesidades del cliente aplicando las pruebas de caja negra en el componente

Con el objetivo de validar que la solución satisfaga las necesidades del cliente, en el presente Trabajo de Diploma se aplicó la prueba de Caja negra. Para ello se emplearon las técnicas de Partición de equivalencia y Análisis de valores límites, que se aplicaron a los requisitos funcionales. Las variables de equivalencia representan un conjunto de estados válidos y no válidos para las condiciones de entrada de un sistema. La aplicación de estas técnicas a los requisitos funcionales definidos,

permitieron señalar que los resultados obtenidos son satisfactorios desde el punto de vista funcional del componente, atendiendo al correcto comportamiento del mismo ante diferentes situaciones de entradas válidas y no válidas. Estos resultados se pudieron comprobar haciendo uso de los Casos de prueba, donde al aplicarlos se pudo obtener un resultado que prueba que el componente presenta un correcto funcionamiento de acuerdo con las necesidades del cliente.

Diseño de los casos de prueba

Unas de las formas para comprobar los requisitos de software es utilizando los diseños de Casos de prueba para efectuar las pruebas de Caja negra. En este caso, se intenta encontrar el mayor número de errores con la menor cantidad de esfuerzo y tiempo, según lo indicado por (S. Pressman, 2001). A continuación se representa el diseño de caso de prueba para el requisito funcional Configurar Análisis Financiero tomando el flujo alternativo para cambios en pesos y porcentajes, los diseños de casos de pruebas restantes se encuentran en el expediente de proyecto del centro CEIGE, además de los restantes artefactos entregados del componente.

Diseño de caso de prueba Requisito Funcional. Configurar Análisis Financiero **Condiciones de ejecución**

- ✓ Se debe identificar y autenticar ante el sistema y además debe tener los permisos para ejecutar esta acción.
- ✓ Se debe seleccionar el subsistema de **Contabilidad**.
- ✓ Se debe seleccionar la opción **Contabilidad Financiera/Análisis Financiero/Análisis Financiero**.
- ✓ Debe existir al menos un estado financiero registrado.

Requisitos a probar

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
----------------------	---------------------	-----------------------	---------------------

<p>1. Configurar Análisis Financiero. Flujo alternativo Cambio en pesos y porcentajes.</p>	<p>Se realiza el análisis por la técnica Cambio en Pesos y Porcentajes.</p>	<p>EP 1.1: Realizar Técnica introduciendo los datos válidos.</p>	<ul style="list-style-type: none"> - Se carga la interfaz Análisis Financiero. - Se selecciona una de las dos opciones por la que se realizara el análisis, por período o ejercicio. - Si se selecciona el RadioButton Ejercicio, solo deberá quedar habilitada la lista desplegable correspondiente al ejercicio base y deshabilitada además la lista desplegable de selección de ejercicios a comparar y la de período base. - Si se selecciona el RadioButton Período, se deberá habilitar la lista desplegable correspondiente al ejercicio base. Una vez elegida el ejercicio se habilita la lista para seleccionar el período correspondiente a ese ejercicio. - Se selecciona el intervalo de períodos o ejercicios a comparar, los cuales no pueden ser mayor que 3.
--	---	--	---

			<ul style="list-style-type: none"> - Se selecciona además el estado financiero a analizar. - Se selecciona la técnica Cambio en pesos y porcentajes. - Se presiona el botón Realizar Análisis.
		EP 1.2: Realizar Técnica seleccionando datos idénticos.	<ul style="list-style-type: none"> - Se seleccionan los datos anteriormente expuestos. - Se presiona el botón Realizar Análisis. - El sistema muestra un mensaje indicando la selección de datos idénticos, en este caso los datos serían: período o ejercicio base igual a ejercicio o período a comparar. - Se presiona el botón Aceptar del mensaje.
		EP 1.3: Realizar Técnica dejando campos sin seleccionar.	<ul style="list-style-type: none"> - Se dejan campos sin seleccionar. - Se presiona el botón Realizar Análisis. - Se muestra un mensaje de error informando los campos vacíos de selección obligatoria. - Se presiona el botón Aceptar del mensaje de error.

Tabla 12 Descripción caso de prueba Configurar Análisis Financiero.

Descripción de variable

No	Nombre de campo	Tipo	Válido	Inválido	Inválido	Inválido
1	Ejercicio o Período	RadioButton.	--	--	--	--
2	Ejercicio o Período base.	Lista Desplegable	Período o ejercicio	Vacío	--	--
3	Ejercicios o Períodos a comparar	ItemSelector	Períodos o ejercicios	Vacío	4 o más	
4	Estados Financieros	GridPanel	Estado situación o Estados Resultados.	Vacío	--	--
5	Tipo de técnica	Lista Desplegable	Cambio en pesos y porcentajes.	Vacío	Otra selección	--

Tabla 13 Descripción de variable.

Juegos de datos a probar

Id	Escenario	Ejercicio/Período	Ejercicio/Período base	Ejercicios/Períodos a comparar	Estados Financieros	Tipo de técnica	Respuesta del sistema
EP 1.1	Realizar Técnica introduciendo los datos válidos.	V(Seleccionar)	V(Seleccionar)	V(Períodos o Ejercicios)	V(Seleccionar)	V(Seleccionar)	El sistema debe permitir realizar el análisis y mostrar un reporte de dicha técnica seleccionada, dando la opción de Imprimir.
EP 1.2	Realizar Técnica seleccionando datos idénticos.	V(Seleccionar)	V(Seleccionar)	V(Períodos o Ejercicios)	V(Seleccionar)	V(Seleccionar)	El sistema debe mostrar un mensaje indicándole al usuario los datos idénticos, permitiendo rectificar el error.

EP 1.3	Realizar Técnica dejando campos sin seleccionar.	I(Vacío)	I(Vacío)	I(Vacío)	I(Vacío)	I(Vacío)	El sistema debe mostrar mensajes en dependencia de los campos que no se seleccionó.
--------	--	----------	----------	----------	----------	----------	---

Tabla 14 Juego de datos a probar.

Resultados de las pruebas

Fue realizada la validación del componente desarrollado aplicándose los distintos casos de prueba donde se detectaron en total 6 no conformidades (NC) de ellas fueron 3 NC en la primera iteración de las pruebas y para la segunda iteración se detectaron 3 nuevas NC que fueron resueltas para la tercera iteración en la cual no fueron encontradas NC. De esta forma se cumplió con el objetivo principal de las pruebas al asegurar que el componente cumple con las especificaciones requeridas y la eliminación de los posibles defectos que este pudiera tener. Por tanto se obtiene el Acta de Liberación por parte del Departamento de Calidad del Centro CEIGE, con la firma de la Administradora de Calidad, la Ing. Giselle Almeida González y de la jefa de la línea Contabilidad la Ing. Yanay Hernández Sosa.

3.5. Conclusiones parciales

Al finalizar el presente capítulo vale destacar que:

- Con la validación del modelo de negocio, se posibilitó comprobar que las técnicas utilizadas para descubrir el negocio fueron aplicadas de forma correcta, además de que su descripción se realizó satisfactoriamente, siendo fieles a las necesidades del cliente.
- La utilización de las técnicas definidas para la validación de requisitos, demostró que estos presentan las condiciones requeridas, y permitió continuar con el diseño e implementación de los mismos, logrando la entrega del producto en tiempo y en correspondencia con las necesidades del cliente.
- Con la aplicación de las métricas destinadas a validar el diseño se pudo constatar que este posee una buena calidad, posibilitando una futura reutilización del código y de las clases en diseños futuros.

- Se obtienen importantes resultados sobre el comportamiento del componente aplicando las pruebas de Caja negra. Estas permitieron comprobar los requisitos funcionales definidos para el componente a partir de los diseños de casos de pruebas, evaluándose la calidad del mismo para lograr satisfacer las necesidades del cliente.

CONCLUSIONES

Una vez terminado el presente Trabajo de Diploma se puede concluir que se desarrollaron todas las tareas a fin de cumplir los objetivos propuestos, resaltando que:

- ✓ Después del análisis de los sistemas estudiados, se pudo obtener información relacionada con la realización del análisis financiero, permitiendo emplear algunas de sus interfaces de usuario y formatos para mostrar la información como guía para la implementación del componente como propuesta de solución.
- ✓ La utilización de las técnicas para la captura de los requisitos, permitieron identificar los requisitos funcionales del componente, que de conjunto a la aplicación de las etapas de la Ingeniería de Requisitos desarrollaron y administraron los mismos durante la investigación.
- ✓ El desarrollo de la solución posibilitó la realización de los análisis financieros a los estados financieros, satisfaciendo las necesidades del cliente y sirviendo de apoyo al proceso de toma de decisiones en las entidades empresariales y presupuestadas.
- ✓ Las pruebas de Caja negra permitieron encontrar y corregir los errores no detectados durante la implementación posibilitando cumplir con las especificaciones requeridas y la validación del componente implementado.

Por tanto, al finalizar el presente trabajo, queda informatizada la realización de los análisis financieros a los estados financieros, en correspondencia con los requerimientos definidos, alcanzándose de esta manera los objetivos propuestos para dar solución al problema planteado. El desarrollo de esta aplicación tiene gran importancia, porque contribuye al mejoramiento del trabajo de los contadores, aportando un poco más a las entidades empresariales y presupuestadas u órgano financiero a escala nacional donde se requiera realizar análisis que brinden información financiera.

RECOMENDACIONES

- ✓ Se recomienda la constante actualización del componente para futuras transformaciones en la medida en que las necesidades de los clientes puedan ser modificadas por el avance de la ciencia Contable y del mundo en general.
- ✓ Utilizar las tecnologías y herramientas estudiadas para la realización de este componente, en el desarrollo de otros sistemas informáticos relacionados con la gestión de los procesos contables.

BIBLIOGRAFÍA

Andux, Yadira Piñera. 2010. Formalización y estandarización de la documentación técnica de la arquitectura tecnológica del Marco de Trabajo Sauxe versión 2.0. La Habana, Cuba : UCI, 2010.

Arizaca, Lic. Eliza. 2009. *Análisis y diseño de sistemas II*. La Paz, Bolivia : s.n., 2009.

ASSETS. 2012. ASSETS Sistema de gesti'on integral. [En línea] 2012. [Citado el: 17 de Enero de 2012.] <http://www.assets.co.cu/ventajas.asp>.

Bernal, Ing. Annilie Manresa. 2009. *Glosario de términos de Contabilidad General 1.0*. CedruX, Universidad de las Ciencias Informáticas. 2009.

Bruzón Estrada, Joiser. 2010. *Especificación de la arquitectura del sistema Contabilidad y Finanzas*. 2010.

BSE consulting. BSE it solution consulting. [En línea] [Citado el: 5 de 2 de 2012.] <http://www.bse-c.co.kr/en/products/microsoft-dynamics-nav>.

Carnegie Mellon University. 2012. Software Engineering Institute. [En línea] 2012. [Citado el: 17 de Enero de 2012.] <http://www.sei.cmu.edu/architecture/start/glossary/community.cfm>.

Casanovas, Josep. 2004. desarrolloweb.com. [En línea] 9 de 9 de 2004. [Citado el: 23 de 11 de 2012.] <http://www.desarrolloweb.com/articulos/1622.php>.

CENDITEL. 2011. Plataforma de Desarrollo de Software Libre (PDSL). *Manual del Usuario del Sistema de Control de Versiones (SVN)*. [En línea] CENDITEL, 2011. [Citado el: 20 de 11 de 2012.] <http://plataforma.cenditel.gob.ve/wiki/ManualUsuarioSvn>.

Chile, Universidad de. 2010. Departamento de Ciencias de la Computación. *Sitio Web DCC*. [En línea] 2010. [Citado el: 05 de Junio de 2012.] <http://www.dcc.uchile.cl/~psalinas/uml/modelo.html>.

Círculo de Periodistas Cubanos contra el Terrorismo. 2012. Cubadebate, contra el terrorismo mediático. [En línea] 2012. [Citado el: 14 de Enero de 2012.] http://www.cubadebate.cu/wp-content/uploads/2011/05/tabloide_debate_lineamientos.pdf.

CITMATEL. Rodas XXI. Sistema Integral Económico Administrativo. [En línea] [Citado el: 26 de 11 de 2012.] <http://www.rodasxxi.cu/rodasxxi.php>.

Clements, Paul y Northrop, Linda. 1996. Software architecture: An executive overview. [En línea] Febrero de 1996. [Citado el: 17 de Enero de 2012.] <http://www.sei.cmu.edu/library/abstracts/reports/96tr003.cfm>. Technical Report CMU/SEI-96-TR-003.

Colectivo de autores: UCID. 2009. *Proceso de Desarrollo y Gestión de Proyectos de Software*. Ciudad de la Habana : s.n., 2009.

Colectivo de JSON. 2011. JSON. [En línea] JSON, 2011. [Citado el: 30 de Noviembre de 2011.] <http://www.json.org/json-es.html>.

Colectivo editorial W3C Consortium. 2008. W3C Consortium. [En línea] 9 de 1 de 2008. [Citado el: 19 de 11 de 2012.] <http://www.w3c.es/Divulgacion/GuiasBreves/TecnologiasXML>.

Companioni, DrC. Andrés Fernández. 2004. *Diccionario de Términos Económicos, Agroeconómicos y Contables*. 2004.

de la Paz Machín, Adolfo. 2011. *Sistema de cuadros financieros del subsistema de Contabilidad General*. 2011.

del Toro Ríos, Dr. C José Carlos and González Brito, Ing. Henry Raúl. 2008. *Documento Visión ERP Cuba*. 2008.

Dirección de Economía del MINFAR. 2003. *Manual del Sistema de Contabilidad Financiera para la Actividad Presupuestada en las FAR*. Plaza de la Revolución, Ciudad de la Habana : Centro de Información para la Defensa del MINFAR, 2003.

Doctrine-Project.org. 2010. Doctrine. [En línea] 2010. [Citado el: 23 de 11 de 2012.] <http://www.doctrine-project.org>.

E.U. Ingeniería Técnica Informática de Oviedo. Sitio web de la E.U.I.T.I.O. [En línea] [Citado el: 5 de Noviembre de 2011.] <http://www.petra.euitio.uniovi.es%2F~i1667065%2FHD%2Fdocumentos%2FEntornos%2520de%2520Desarrollo%2520Integrado.pdf&ei=GannTrWxL8Pqgge-2>.

eco-finanzas.com. eco-finanzas.com. *ESTADO DE RESULTADOS*. [En línea] [Citado el: 18 de 11 de 2012.] http://www.eco-finanzas.com/diccionario/E/ESTADO_DE_RESULTADOS.htm.

ERP Access. 2009. ERP Access. [En línea] 2009. [Citado el: 5 de 2 de 2013.] <http://erpaccess.blogspot.com/2012/11/analisis-financiero-con-microsoft-access.html>.

Escalona, María José y Koch, Nora. 2002. *Ingeniería de Requisitos en Aplicaciones para la Web*. Sevilla : s.n., 2002.

Eslomas.com. 2007. Eslomas.com. *Frameworks de Zend para el desarrollo de aplicaciones PHP*. [En línea] 3 de 7 de 2007. [Citado el: 23 de 11 de 2012.] <http://www.eslomas.com/index.php/archives/2007/07/03/framework-de-zend-para-el-desarrollo-de-aplicaciones-php/>.

Estrada, 1er Tte. Joiser Bruzón. 2009. *Especificaciones complementarias*. Línea de Contabilidad y Finanzas, Unidad de Compatibilización, Integración y Desarrollo de software para la defensa. 2009.

Fowler, Martin. 2003. *Patterns of enterprise application architecture*. s.l. : Addison-Wesley, 2003. ISBN 978-0-321-12742-6.

Gómez Baryolo, Oiner, Morejón Borbón, Yoandry y García Tejo, Darien. 2012. *Arquitectura tecnológica para el desarrollo de software*. 2012.

González, Mairelys Fernández y Rivera, Osley Zorrilla. 2010. Diseño e implementación del componente Ajuste al Costo del Subsistema Costos y Procesos del Sistema Integral de Gestión de Entidades CEDRUX. *Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas*. Ciudad de la Habana : UCI, 2010.

Informática Hoy. Informática Hoy. [En línea] [Citado el: 27 de 11 de 2012.] <http://www.informatica-hoy.com.ar/sap/Que-es-SAP.php>.

Ingeniería Técnica Informática de Oviedo. 2010. Sitio web de la E.U.I.T.I.O. [En línea] 2010. [Citado el: 20 de 11 de 2012.] <http://www.petra.euitio.uniovi.es%2F~i1667065%2FHD%2Fdocumentos%2FEntornos%2520de%2520Desarrollo%2520Integrado.pdf&ei=GannTrWxL8Pqgge-2>.

Instituto Tecnológico Superior Escársega. 2010. Scribd. [En línea] 6 de 2010. [Citado el: 10 de 11 de 2012.] <http://es.scribd.com/doc/37187866/Requerimientos-funcionales-y-no-funcionales>.

Larman, Craig. 1999. *UML y Patrones Introducción al análisis y diseño orientado a objetos*. México : Prentice Hall, 1999. ISBN 0-13-748880-7.

Layola, William. 2006. *Maestría en Sistemas de Información Gerencial*. 2006.

Leon, Alexis. 2008. *Enterprise Resources Planning*. New Delhi : Tata McGraw-Hill Publishing Company Limited, 2008.

MARCELO DE LA CRUZ H, CARLOS PALOMINO HURTADO. 2007. *ManualSiscontGold*. 2007.

Martinez, Rafael. 2009. PostgreSQL-es. [En línea] 6 de 6 de 2009. [Citado el: 20 de 11 de 2012.] <http://www.postgresql.org.es/node/297>.

Meltom Technologies. 2005. Gerencia y Negocios en HispanoAmérica. [En línea] 2005. [Citado el: 9 de 10 de 2012.] http://www.degerencia.com/articulo/los_erps_sistemas_integradores.

Metodologías de Sistemas. 2007. Metodologías de Sistemas. [En línea] 2007. [Citado el: 25 de 11 de 2012.] <http://metodologiasdesistemas.blogspot.com/2007/10/que-es-un-orm-object-relational-mapping.html>.

Microsoft Corporation. Septiembre de 2010. *Microsoft Dynamics NAV: Caso de éxito: Unión Gestión Hipotecaria*. Septiembre de 2010.

Microsoft. 2011. Microsoft Dynamics[Soluciones empresariales ERP y CRM. [En línea] Microsoft Dynamics, 2011. [Citado el: 26 de 11 de 2012.] <http://www.microsoft.com/dynamics/es/es/default.aspx>.

MINBAS. 2012. TECNOMATICA. [En línea] 2012. [Citado el: 26 de 11 de 2012.] <http://tecnomatica.minbas.cu/Sistcont.html>.

Ministerio de Finanzas y Precios. Gaceta Oficial de la República de Cuba. *Gaceta Oficial de la República de Cuba*. [En línea] [Citado el: 23 de Enero de 2012.] <http://www.gacetaoficial.cu>. ISSN 1682-7511.

Ministerio de la Informática y las Comunicaciones. 2011. Agencia de Control y Supervisión. [En línea] 2011. [Citado el: 26 de 11 de 2012.] http://www.acs-mic.cu/sw_certificados.htm.

Moreno, Abraham Perdomo. *Análisis e interpretación de estados financieros*. México, DF : Thomson Learning. 9706862633.

Moreno, Adarlis Fernández y Fontanills, Edwing Robert Odelín. 2009. Modelado de negocio y Levantamiento de requisitos del subsistema Activos Fijos Tangibles del sistema CedruX. *Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas*. La Habana : UCI, 2009.

Mozilla Developer Network. 2011. Mozilla Developer Network. [En línea] 09 de 11 de 2011. [Citado el: 5 de 22 de 2013.] https://developer.mozilla.org/es/docs/Gu%C3%ADa_JavaScript_1.5/Concepto_de_JavaScript.

Mozilla Firefox. 2009. GetFirefox. [En línea] 2009. [Citado el: 22 de 11 de 2012.] <http://www.getfirefox.es/firefox-features>.

Nodarse, Dr. Hiram Marquetti. 2007. Cuba siglo XXI. *La Empresa Cubana: Principales Retos que Enfrenta*. [En línea] 2007. [Citado el: 9 de 11 de 2012.] http://www.nodo50.org/cubasigloXXI/economia/marquetti2_310102.htm.

Obregón, Ing. William González. 2012. *Modelo de desarrollo de software*. La Habana, Cuba : UCI, 2012.

Observatorio Tecnológico. 2008. Observatorio Tecnológico. [En línea] 2008. [Citado el: 22 de 5 de 2013.] <http://recursostic.educacion.es/observatorio/web/es/software/servidores/580-elvira-mifsud>.

Olivo De Latouche, Marfa y Maldonado G, Ricardo. 2010. *Estudio de la Contabilidad General*. 2010.

Openbravo S.L. 2012. *Manual de usuario v1.1*. [En línea] 2012. [Citado el: 26 de 11 de 2012.] <http://www.openbravo.com/es>.

OpenERP Spain. openerpspain.com. *OpenERP Spain -Gestión contable y financiera*. [En línea] [Citado el: 5 de 2 de 2013.] <http://www.openerpspain.com/gestion-contable-y-financiera>.

Oracle Corporation. NetBeans. [En línea] [Citado el: 30 de Noviembre de 2011.] <http://netbeans.org/community/releases/70/>.

Parametric Technology Corporation (PTC). 2006. *Verificación y validación*. 2006. 2089-VV-TS-EN1206-ES.

Paz, Alarcón Armenteros y Ulloa. 2012. Revista académica de economía. *Observatorio de la Economía Latinoamericana*. [En línea] 2012. [Citado el: 9 de 10 de 2012.] <http://www.eumed.net/cursecon/ecolat/cu/2012/aaup.html>. ISSN 1696-8352.

Pérez Pérez, Joisel. 2010. *Vista de datos de la arquitectura Contabilidad y Finanzas CONFIN*. 2010.

Pérez, Ing. Joisel Pérez. 2010. *Línea base de la arquitectura de software Contabilidad*. s.l. : ERP Cuba, 2010.

Pérez, Joisel Pérez. 2012. *Solucion para la obtención de los indicadores financieros en CEDRUX*. La Habana, Cuba : s.n., 2012.

Pressman, Roger S. 2005. *Ingeniería del Software. Un enfoque práctico*. 2005.

Proyecto SAUXE. 2012. *Arquitectura de Software. Vista de integración*. 2012.

Ramírez, Jaime. 2012. *Métodos de Prueba del Software*. Unidad de Programación. [En línea] 2012. [Citado el: 28 de mayo de 2012.] <http://lml.ls.fi.upm.es/>.

Resolución No. 235-2005, Ministerio de Finanzas y Precios. 2005. *MARCO CONCEPTUAL PARA LA PREPARACIÓN, PRESENTACIÓN Y PUBLICACIÓN DE LOS ESTADOS FINANCIEROS*. Ciudad de La Habana : s.n., 2005.

RodasXXI. rodasxxi.cu. [En línea] RodasXXI. [Citado el: 28 de 11 de 2012.] <http://www.rodasxxi.cu/contabilidad.php>.

Romero, Omar Casasola. 2010. *Programacion en Castellano*. [En línea] 27 de Agosto de 2010. [Citado el: 30 de Noviembre de 2011.] http://www.programacion.com/articulo/introduccion_a_uml_181.

S. Pressman, Roger. 2001. *Ingeniería de Software un enfoque práctico*. Madrid : s.n., 2001.

Sanchez, Saul Alcalá. 2006. *GestioPolis*. [En línea] 02 de 2006. [Citado el: 9 de 10 de 2012.] http://www.gestiopolis.com/dirgp/fin/analisis_financiero/analisis_3.htm.

SAP AG. 2010. *SAP Business One. Gestión financiera con SAP® Business One: Gestión de la contabilidad, flujo de caja y las operaciones bancarias*. [En línea] 2010. [Citado el: 27 de 11 de 2012.]

http://www.sap.com/spain/sme/solutions/businessmanagement/businessone/pdf/6313_PB_50087305_esES.pdf.

Scribd. Scribd. [En línea] [Citado el: 20 de 11 de 2012.] <http://www.scribd.com/doc/3062020/Capitulo-I-HERRAMIENTAS-CASE>.

SINCA - Financiero - Microsoft Dynamics GP / Microsoft Dynamics CRM. [En línea] [Citado el: 18 de 11 de 2012.] <http://www.sinca.com/Soluciones/ERP/Financiero/tabid/88/Default.aspx>.

2007. Sparx Systems. [En línea] Sparx Systems Pty Ltd, 2007. [Citado el: 5 de mayo de 2011.] http://www.sparxsystems.com.ar/resources/tutorial/uml2_deployementdiagram.html.

The FreeBSD Foundation. 2012. The FreeBSD Project. [En línea] 2012. [Citado el: 23 de 11 de 2012.] <http://www.freebsd.org/doc/es/articles/explaining-bsd/article.html>

The PHP Group. 2011. PHP: Hipertext Preprocesor. [En línea] php.net, 4 de 2 de 2011. [Citado el: 19 de 11 de 2012.] <http://php.net/manual/es/intro-what-is.php>.

Toro, Amador Durán. 2000. *Un Entorno Metodológico de Ingeniería de Requisitos para Sistemas de Información*. Sevilla : s.n., 2000.

Unidad de Compatibilización Integración y Desarrollo. 2009. *Proceso de Desarrollo y Gestión de Proyectos de Software*. Ciudad de la Habana : s.n., 2009.

Vázquez, Lic. Rafael Cisneros y Labrada, Lic. Alexander Espinoza. 2010. Ilustrados. [En línea] 2010. [Citado el: 11 de 10 de 2012.] <http://www.ilustrados.com/tema/13259/Analisis-interpretacion-estados-financieros.html>.

Visual Paradim. freedownloadmanager.org. [En línea] freedownloadmanager.org. [Citado el: 30 de Noviembre de 2011.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/.

wiki.openbravo.com. 2012. wiki.openbravo.com. [En línea] 3 de 4 de 2012. [Citado el: 27 de 11 de 2012.] http://wiki.openbravo.com/wiki/ERP_2.50:Functional_Description/Financial_management/es#Herramientas_de_an.C3.A1lisis_2.

GLOSARIO DE TÉRMINOS

Ant: Es una herramienta Open-Source utilizada en la compilación y creación de programas Java, Está escrito en XML y Java.

Refactoring: Es una técnica disciplinada para la reestructuración de un cuerpo existente de código, alterando su estructura interna sin cambiar su comportamiento externo.

BSD: La licencia BSD (Berkeley Software Distribution) fue creada inicialmente para los sistemas operativos de la Universidad de Berkeley. Se califica como una licencia mucho más libre que la GPL y más rápida.

Log: Registro oficial de eventos durante un rango de tiempo en particular. Para los profesionales en seguridad informática es usado para registrar datos o información sobre: quien, que, cuando, donde y por qué: un evento ocurre. Por lo tanto se puede decir que un log es una evidencia digital.

Secure Sockets Layer: protocolo criptográfico que proporciona comunicación segura en Internet.

ExtJS: Es una librería construida con JavaScript para la construcción de componentes para el diseño de interfaces de usuario del lado del cliente haciendo uso extensivo de Ajax. Emplea una arquitectura flexible que permite construir aplicaciones complejas utilizando componentes predefinidos.

Doctrine: Es un mapeador objetos relacional de para PHP. Permite trabajar con los datos persistidos como si fueran parte de una base de datos orientada a objetos posibilitando escribir consultas a la base de datos en un dialecto orientado a objetos de propiedad SQL.

Zend Framework: Es un Framework para el desarrollo de aplicaciones y servicios Web con PHP. Es de código abierto y emplea PHP 5 en sus más recientes versiones.

DDL (Data Definition Language): es la parte del SQL dedicada a la definición de la base de datos, consta de sentencias para definir la estructura de esta, permitiendo especificar gran parte de su nivel interno.

Anexo 2 Prototipos de interfaz de usuario funcional

Partidas	Valor en pesos			2019 / 2011		2018 / 2011	
	2019	2018	2011	Valor	%	Valor	%
Nombre partidas							
Cuentas por cobrar a largo plazo	0.00	0.00	7272874.00	(7272874)	(100)	(7272874)	(100)
Cuentas por cobrar a largo plazo	0.00	0.00	7272874.00	(7272874)	(100)	(7272874)	(100)
Cuentas por cobrar a largo plazo	0.00	0.00	7272874.00	(7272874)	(100)	(7272874)	(100)
Grupo de activos a largo plazo	0.00	0.00	7272874.00	(7272874)	(100)	(7272874)	(100)
Efectos por cobrar	0.00	0.00	544558.00	(544558)	(100)	(544558)	(100)
Cuentas por cobrar	0.00	0.00	4310652.00	(4310652)	(100)	(4310652)	(100)
Pagos anticipados a suministradores	0.00	0.00	263918.00	(263918)	(100)	(263918)	(100)
Adeudos del presupuesto del estado	0.00	0.00	627180.00	(627180)	(100)	(627180)	(100)
Adeudos del organismo	0.00	0.00	2.00	(2)	(100)	(2)	(100)
Materias primas y materiales	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Partes y Piezas de Repuesto	0.00	0.00	4744.00	(4744)	(100)	(4744)	(100)
Utiles y Herramientas	0.00	0.00	16522.00	(16522)	(100)	(16522)	(100)
Mercancias para la venta	0.00	0.00	5273886.00	(5273886)	(100)	(5273886)	(100)
Inventarios Ociosos	0.00	0.00	1339468.00	(1339468)	(100)	(1339468)	(100)
Inventarios	0.00	0.00	157102.00	(157102)	(100)	(157102)	(100)
Efectivo en caja y banco	0.00	0.00	616354.00	(616354)	(100)	(616354)	(100)
Grupo de activos circulantes	0.00	0.00	13154386.00	(13154386)	(100)	(13154386)	(100)
Activos fijos tangibles	0.00	0.00	2559938.00	(2559938)	(100)	(2559938)	(100)
Depreciación de activos fijos tangibles	0.00	0.00	1016190.00	(1016190)	(100)	(1016190)	(100)
Activos fijos tangibles en ejecución	0.00	0.00	783114.00	(783114)	(100)	(783114)	(100)
Grupo de activos fijos tangibles	0.00	0.00	2326862.00	(2326862)	(100)	(2326862)	(100)
Gastos diferidos a largo plazo	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Figura 18 Cambios en pesos y porcentajes.

Partidas	2019		2018		2011	
	Valor 2019	%	Valor 2018	%	Valor 2011	%
Nombre partidas						
Cuentas por cobrar a largo plazo	0.00	0.00	0.00	0.00	7272874.00	100.00
Cuentas por cobrar a largo plazo	0.00	0.00	0.00	0.00	7272874.00	100.00
Cuentas por cobrar a largo plazo	0.00	0.00	0.00	0.00	7272874.00	100.00
Grupo de activos a largo plazo	0.00	0.00	0.00	0.00	7272874.00	(33276.33)
Efectos por cobrar	0.00	0.00	0.00	0.00	544558.00	4.14
Cuentas por cobrar	0.00	0.00	0.00	0.00	4310652.00	32.77
Pagos anticipados a suministrad...	0.00	0.00	0.00	0.00	263918.00	2.01
Adeudos del presupuesto del est...	0.00	0.00	0.00	0.00	627180.00	4.77
Adeudos del organismo	0.00	0.00	0.00	0.00	2.00	0.00
Materias primas y materiales	0.00	0.00	0.00	0.00	0.00	0.00
Partes y Piezas de Repuesto	0.00	0.00	0.00	0.00	4744.00	0.04
Utiles y Herramientas	0.00	0.00	0.00	0.00	16522.00	0.13
Mercancias para la venta	0.00	0.00	0.00	0.00	5273886.00	40.09
Inventarios Ociosos	0.00	0.00	0.00	0.00	1339468.00	10.18
Inventarios	0.00	0.00	0.00	0.00	157102.00	1.19
Efectivo en caja y banco	0.00	0.00	0.00	0.00	616354.00	4.69
Grupo de activos circulantes	0.00	0.00	0.00	0.00	13154386.00	(60186.61)
Activos fijos tangibles	0.00	0.00	0.00	0.00	2559938.00	110.02
Depreciación de activos fijos tan...	0.00	0.00	0.00	0.00	1016190.00	43.67
Activos fijos tangibles en ejecución	0.00	0.00	0.00	0.00	783114.00	33.66
Grupo de activos fijos tangibles	0.00	0.00	0.00	0.00	2326862.00	(10646.33)
Gastos diferidos a largo plazo	0.00	0.00	0.00	0.00	0.00	0.00

Figura 19 Porcentajes componentes.

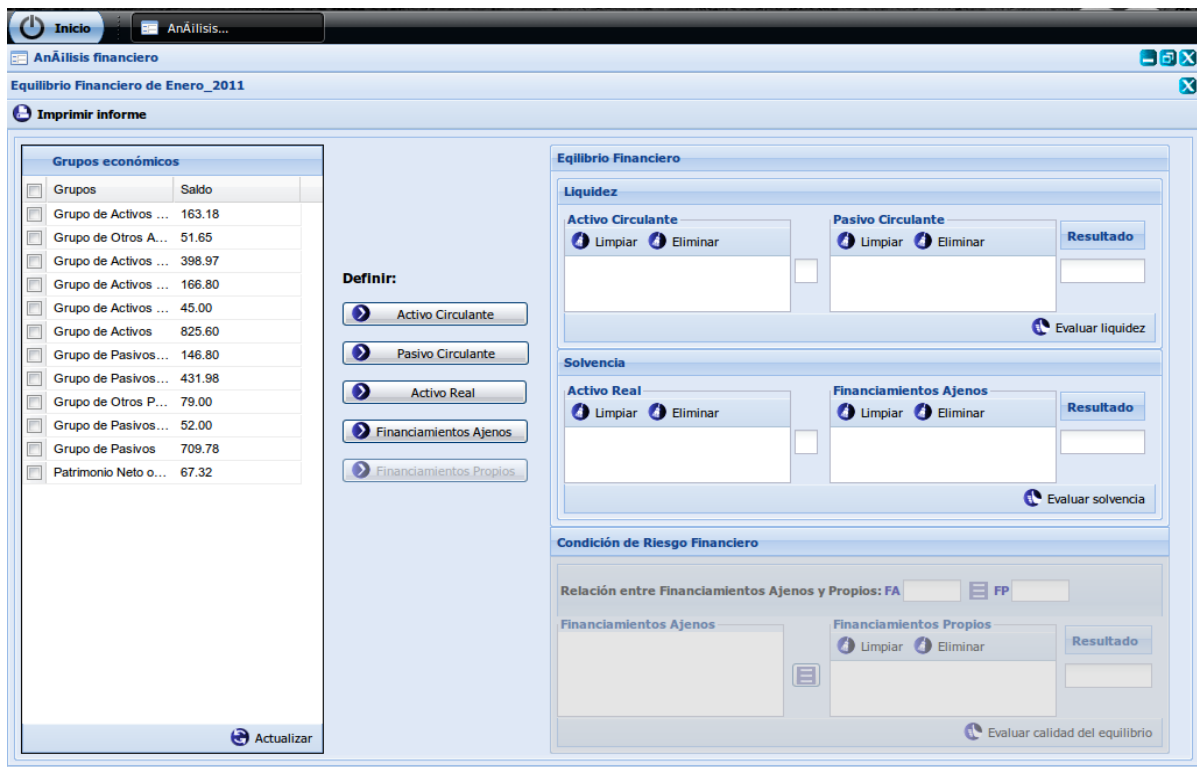


Figura 20 Equilibrio Financiero.

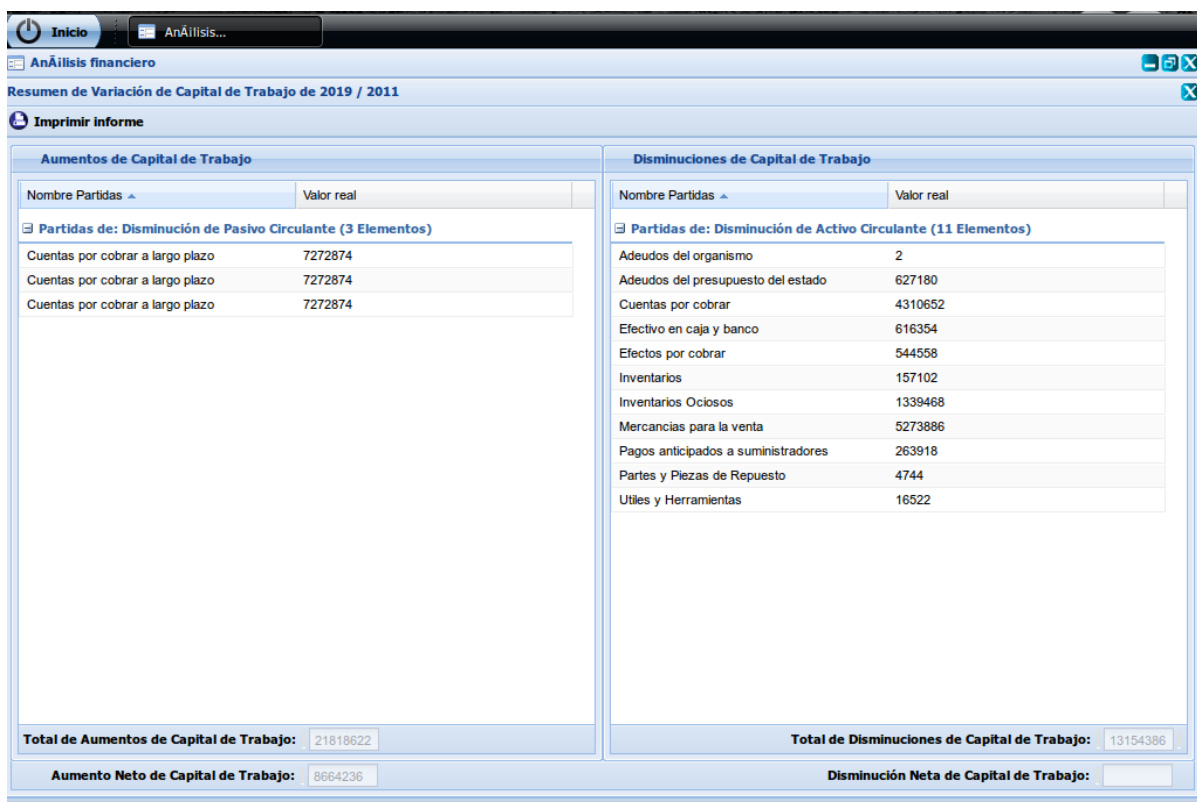


Figura 21 Variación del capital de trabajo



Figura 22 Graficar Estados Financieros: Línea

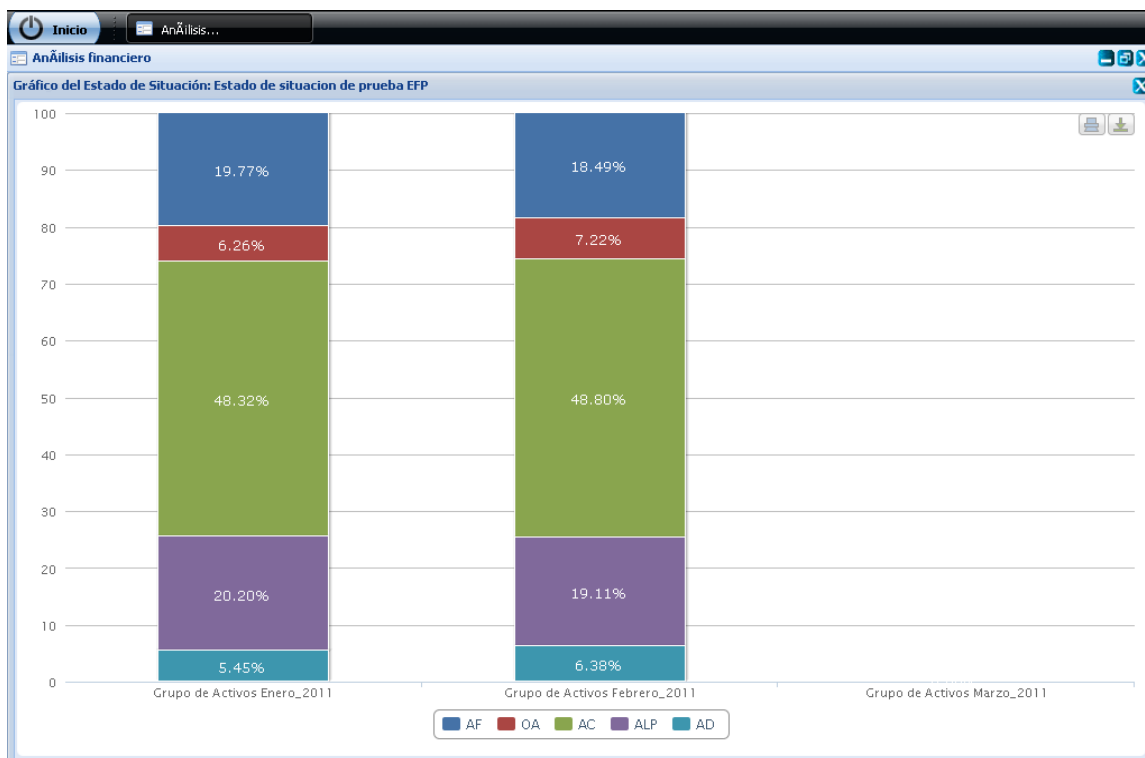


Figura 23 Graficar Estados Financieros: Barra