

**FACULTAD DE INGENIERÍA
INGENIERÍA INFORMÁTICA**



TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERÍA EN INFORMÁTICA

TÍTULO:

“SISTEMA PARA CONTRIBUIR AL APOYO DOCENTE DE LA ASIGNATURA ESTUDIO Y ANÁLISIS DE LOS ESTADOS FINANCIEROS DE LA CARRERA CONTABILIDAD Y FINANZAS EN LA UNIVERSIDAD DE SANCTI SPÍRITUS JOSÉ MARTÍ PÉREZ”

AUTOR:

Jorge Luis Pardillo Castañeda.

TUTORES:

Ing. Arley Ulloa Zaila.

Lic. Yelianny Hernández González

SANCTI SPÍRITUS-2015

DEDICATORIA

- ✚ A mi Hermana y mis Padres que me han apoyado durante toda mi vida.
- ✚ A mi novia por estar a mi lado en todo momento.
- ✚ A mis tutores por brindarme toda la ayuda y el apoyo necesario.
- ✚ A todos mis amigos.

AGRADECIMIENTOS

- ✚ A mi hermana Martha Karina por apoyarme siempre en todo momento.
- ✚ A mis padres Jorge Luis y Olga Nila por estar siempre a mi lado guiándome y mostrándome el camino que hay que seguir.
- ✚ A mi novia Caridad Osorio por ese apoyo incondicional en todo momento y por estar siempre a mi lado.
- ✚ A mis tutores Arley y Yelianny por enseñarme los pasos necesarios para la realización de esta tesis.
- ✚ A Maritza y el Chino por ese apoyo brindado en todo momento.
- ✚ A mis amigos Frank, Irving, Alejandro y Gabriel.

RESUMEN

Este trabajo se desarrolla como una herramienta de apoyo docente al proceso de enseñanza - aprendizaje de la asignatura Estudio y Análisis de los Estados Financieros en la carrera de Contabilidad y Finanzas de la Universidad de Sancti Spíritus “José Martí Pérez”, donde actualmente las clases prácticas no se aprovechan al máximo producto de los múltiples cálculos que necesitan realizar los estudiantes para poder interpretar las razones financieras con vista a la toma de decisiones. A pesar de que la carrera tiene a su disposición el sistema VERSAT Sarasola, este, no brinda dichos análisis financieros. Es por ello que el objetivo de esta investigación es desarrollar un sistema informático para contribuir al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez”.

Las etapas llevadas a cabo en la elaboración del software fueron las planteadas por el Proceso Unificado de Desarrollo (RUP), utilizándose como lenguaje al “Lenguaje Unificado de Modelado” (UML) y como herramienta para el análisis y diseño: el *Visual Paradigm For UML*. Para implementar el sistema se utilizó Java como lenguaje de programación lo que proporcionó un código eficiente, como *framework* se utilizó el ORM *Hibernate*, para la persistencia de los datos se utilizó *SQLite Manager 3.5y* como herramienta de programación el *NetBeans* versión 6.9.1 con *plugin* integrado para el trabajo con el reporteador *iReport - JasperReport*.

TABLA DE CONTENIDO

ÍNDICE DE FIGURAS	7
ÍNDICE DE Tablas	8
Introducción	9
Capítulo I: Fundamentos teóricos y metodológicos sobre la creación de un sistema portable para el apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros para la carrera Contabilidad y Finanzas.	16
1.1- Los Estados Financieros	16
1.2- Análisis de las Razones Financieras	17
1.3- Sistema económico integrado VERSAT Sarasola	18
1.4- Metodología utilizada.....	20
1.4.1- Lenguaje Unificado de Modelado.....	20
1.4.2- Proceso Unificado de Desarrollo	21
1.5- Tecnologías y lenguajes utilizados.....	21
1.5.1- El lenguaje de programación Java	22
1.5.2- El ORM Hibernate	23
1.6- Herramientas de desarrollo utilizadas.....	24
1.6.1- El ambiente de desarrollo de NetBeans	24
1.6.2- El sistema de gestión de base de datos SQLite Manager 3.5.....	25
1.6.3- El generador de reportes iReport - JasperReport.....	25
1.6.4- El CASE de modelado Visual Paradigm For UML	26
Capítulo II: Descripción del sistema propuesto para contribuir al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez”	29
2.1- Modelo del negocio.....	29
2.1.1- Reglas del Negocio	29
2.1.2- Modelo de casos de uso del negocio	30
2.1.3- Actores del negocio	30
2.1.4- Diagrama de casos de uso del negocio	30
2.1.5- Trabajadores del negocio	31
2.1.6- Diagramas de actividades.....	31
2.1.7- Modelo de objetos del negocio.....	31

2.2- Requerimientos funcionales.....	32
2.3- Requerimientos no funcionales.....	36
2.4- Modelo de casos de uso del sistema.....	38
2.4.1- Actores del sistema	38
2.4.2- Casos de uso del sistema.....	39
2.4.3- Diagramas de casos de uso del sistema.	40
Capítulo III: Construcción del sistema propuesto para contribuir al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez”	44
3.1- Diseño de clases persistentes.....	44
3.2- Modelo de Datos	44
3.3- Diseño de clases del Sistema.....	45
3.4- Principios de Diseño	45
3.4.1- Diseño de la interfaz del sistema	45
3.4.2- Tratamiento de errores	46
3.4.3- Concepción general de la ayuda	46
3.4.4- Estándares de codificación.....	46
3.5- Modelo de implementación	47
3.5.1- Diagrama de Despliegue.....	47
3.5.2- Diagrama de Componentes.....	47
3.6- Pruebas.....	49
3.6.1- Planificación de prueba	49
3.6.2- Procedimientos de prueba	50
3.6.3- Casos de prueba	51
CONCLUSIONES	53
RECOMENDACIONES	54
BIBLIOGRAFÍA	55
ANEXOS.....	56

ÍNDICE DE FIGURAS

Figura 1 : Diagrama de casos de uso del negocio.....	31
Figura 2 : Modelo de Objetos del Negocio.....	32
Figura 3 : Casos de uso por paquetes.....	41
Figura 4 : Diagrama casos de uso del Sistema Paquete Operaciones.....	42
Figura 5: Diagrama casos de uso del Sistema Paquete Reportes.....	43
Figura 6: Modelo de Datos.....	45
Figura 7: Diagrama de despliegue.....	47
Figura 8: Diagrama de componentes.....	48

ÍNDICE DE TABLAS

Tabla 1: Actores del negocio	30
Tabla 2: Trabajadores del negocio.....	31
Tabla 3: Actores del sistema.....	38
Tabla 4: Descripción de los paquetes	40
Tabla 5: Descripción de los componentes SistApoyo.jar.....	48
Tabla 6: Caso de prueba para el caso de uso Mostrar Razones de Rotación.....	51
Tabla 7: Caso de prueba para el caso de uso Gestionar Activos Circulantes.....	51

INTRODUCCIÓN

La contabilidad es una ciencia que se ocupa de registrar, clasificar y resumir las operaciones mercantiles de un negocio con el fin de interpretar sus resultados. El origen de la contabilidad es de hace muchos años, desde que el hombre comprende que su memoria no era suficiente para guardar la información necesaria. Desde el año 6000 A.C. ya existían los elementos necesarios para la actividad contable. En Grecia, Egipto y en los Valles de Mesopotamia llevaban registros y operaciones financieras de las empresas privadas y públicas en tablillas de barro.

El objetivo de la contabilidad es proporcionar información a los dueños y socios de un negocio sobre lo que se debe y se tiene. Se puede dividir en objetivo administrativo y financiero. El objetivo administrativo es proporcionar información a los administradores para que ellos planifiquen, tomen decisiones, control de las operaciones mientras que el objetivo financiero, por su parte, es proporcionar información de las operaciones realizadas por un ente, fundamentalmente en el pasado.

Se puede afirmar que la contabilidad es un pilar fundamental de las organizaciones, cuyas actividades inciden directamente en la economía de cualquier nación. Nuestro país no está exento del dominio de dicha ciencia, sobre todo a partir de los cambios ocurridos durante la década de los 90 en cuanto a la reestructuración de la economía y la introducción de nuevas técnicas de Gestión Empresarial, las cuales han contribuido a la reanimación económica de varios sectores; también se han creado nuevas empresas producto de las necesidades de servicios que van surgiendo.

En la actualidad, en todos los sectores de la economía, los dirigentes están sometidos a crecientes exigencias y esfuerzos en busca de elevar la eficiencia, eficacia y competitividad, en particular en el sector empresarial, bajo cambiantes condiciones internas y externas. Todo lo anterior exige que la gestión y dirección empresarial se realice en condiciones totalmente diferentes a las de hace poco años, ya que las circunstancias actuales por las que atraviesa el país, la economía está orientada hacia una total recuperación, para ello la elevación de la eficiencia y la productividad social se deberá convertir en la vía principal para asegurar el desarrollo económico. Por tal razón, el estudio y análisis de los estados financiero, constituye la herramienta necesaria para el correcto funcionamiento de cualquier entidad, dado que posibilita la toma de decisiones en la dirección estratégica de la empresa, con el fin de garantizar el empleo más racional de los recursos materiales, financieros y humanos que disponen.

En tal sentido, la necesidad de disponer de una información contable, eficiente y oportuna resulta imprescindible para dirigir económicamente una entidad; la interpretación de los resultados obtenidos y la proyección en el desempeño futuro de la misma, lo que representa un medio efectivo para la dirección, el control de los recursos y su utilización eficiente, lo que implica que la información a un determinado nivel de dirección sea adecuada a las funciones de éste y se reciba con tiempo suficiente para tomar decisiones y controlar su resultado.

Según el contexto económico actual en nuestro país, resulta necesario que el graduado de contabilidad y finanzas este dotado de conocimientos y habilidades que le permitan enfrentar y dar solución a los múltiples problemas que se puedan presentar en dicho contexto. Siendo de vital importancia haber asimilado los conceptos científico-técnicos de su campo profesional, pero además debe en su formación académica desarrollar las habilidades contempladas en el Plan de Estudio.

En nuestras universidades actualmente se encuentra vigente el Plan de Estudio D, en él se encuentra la asignatura Estudio y Análisis de los Estados Financieros, dicha asignatura es impartida en la carrera de Contabilidad y Finanzas en el tercer año, luego de recibir las asignaturas de Contabilidad General I, II, III, IV y Contabilidad Avanzada, así como de tener nociones recibidas de disciplinas como Finanzas, Economía, Administración y de haber tenido vinculación a nuestras empresas mediante las asignaturas Práctica Profesional I y II, es decir, las asignaturas antes mencionadas constituyen la base para enfrentarse al Estudio y Análisis de los Estados Financieros. Esta asignatura le permite al estudiante completar el estudio del ciclo de la contabilidad, ya que una vez culminado el proceso de registro, clasificación y resumen mediante la aplicación de los principios contables se materializa la fase de estudio y análisis de la información contable con vista a la toma de decisiones.

Constituyen objetivos instructivos de la asignatura los siguientes:

- Aplicar las técnicas de actualización y análisis de la información contenida en los Estados Financieros para que contribuya a la efectividad en la toma de decisiones económicas.
- Explicar el fenómeno de la inflación y su efecto en la información financiera de la empresa.

- Registrar los ajustes de las partidas afectadas por la influencia de la inflación aplicando los métodos de re expresión y el valor actual.
- Clasificar las partidas en monetarias y no monetarias
- Actualizar los Estados Financieros aplicando los métodos de re expresión y valor actual.
- Explicar las técnicas de análisis económico y financiero
- Aplicar razones financieras
- Explicar el método Dupont
- Analizar e interpretar la información que brindan los Estados Financieros mediante el uso de razones financieras
- Valorar los resultados obtenidos del análisis de los estados financieros
- Elaborar el Estado de Flujo de Efectivo

Para alcanzar dichos objetivos los estudiantes hacen uso en los ejercicios orientados en clases los Ratios Financieros, los cuales están divididos en seis grandes grupos: Ratios de Liquidez, Ratios de Solvencia, Ratios de Solidez, Ratios de Endeudamiento, Ratios de Rotación, Ratio de Rentabilidad. Cada uno de estos ratios incluye un grupo de razones financieras necesarias para el análisis e interpretación de la información contable que se hace imprescindible obtener y ordenarla según estipula las normas cubanas de contabilidad a través de informes llamados Estado de Situación y Estado de Resultado. Estos ratios no son más que los indicadores económicos y financieros que se requieren para evaluar la situación de la empresa en tres dimensiones a partir de la información contable.

En la clase el profesor de la asignatura Estudio y Análisis de los Estados Financieros orienta un ejercicio en el cual se presenta la información contable (Estado de Situación y Estado de Resultado) del cierre de dos periodos contables dados para una empresa Z (llamémosle a los periodos, periodo X y periodo Y). El profesor entrega la información contable ordenada, ya que la forma de obtenerla le es conocida de cursos anteriores. Luego el estudiante deberá obtener o calcular cada uno de los ratios financieros para luego realizar el estudio y el análisis de la situación financiera de la empresa Z para posteriormente tomar las decisiones oportunas para la

entidad. Si tenemos en cuenta que un ejercicio, el más general posible, puede tener más de 400 indicadores o cuentas (solo en el estado de situación) y la obtención de los ratios financieros se obtienen a través de fórmulas donde se suman, restan y dividen sumas parciales de ciertas y determinadas cuentas de la información contable, existe una gran probabilidad que los estudiantes cometan errores en dichos cálculos a la hora de obtener los ratios financieros.

Según el instrumento de encuesta aplicado a los estudiantes que reciben dicha asignatura y un instrumento de entrevista aplicada a la profesora de la asignatura se pudieron detectar las siguientes deficiencias.

- Pérdidas de tiempos en las clases prácticas al pasar gran parte en un ejercicio calculando razones financieras, cuyo cálculo no es objetivo de la asignatura, sino que interpretan dichos resultados para la toma de decisiones.
- Incapacidad del sistema económico integrado VERSAT Sarasola al no poder obtener las razones financieras, ya que dicho sistema no obtiene la información contable en términos de Estado de Situación y Estado de Resultado, solo llega a obtener el balance de comprobación general.
- No disponibilidad de herramientas solamente especializadas en análisis financiero.
- Inconsistencia en los resultados de las razones financieras obtenidas por los estudiantes producto al factor de error humano al existir muchos cálculos.

Toda esta situación conduce a la existencia del siguiente **problema de investigación** ¿cómo contribuir al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spiritus “José Martí Pérez”?

Para dar solución al problema planteado se toma como **objeto de estudio**: El estudio y análisis de los estados financieros.

Objetivo general de la investigación: Desarrollar un sistema informático portable para contribuir al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spiritus “José Martí Pérez”.

Como **campo de acción**: Obtención de las razones financieras utilizando los estados financieros y el método de análisis horizontal.

Para cumplir el objetivo general se realizan las siguientes **preguntas científicas**:

1. ¿Qué fundamentos teóricos y metodológicos sustentan la elaboración de un sistema informático para apoyar el estudio y análisis de los estados financieros?
2. ¿Cómo diseñar un sistema informático portable para contribuir al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez”?
3. ¿Cómo implementar un sistema informático portable para contribuir al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez”?

En consecuencia con las preguntas científicas se plantean las siguientes **tareas de investigación**:

1. Determinar los fundamentos teóricos y metodológicos que sustenten la elaboración de un sistema informático para apoyar el estudio y análisis de los estados financieros.
2. Diseñar un sistema informático portable que contribuya al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez”.
3. Implementar un sistema informático portable que contribuya al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez”.

La **novedad científica** que aporta la presente investigación radica en el diseño e implementación de un sistema informático portable que permita ser una herramienta de apoyo docente a la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez, dicho sistema permitirá aprovechar el tiempo en las clases prácticas, como una acción necesaria para desarrollar las habilidades necesarias en los estudiantes para la toma de decisiones a partir del método horizontal.

Los métodos utilizados para sustentar científicamente los tres capítulos que componen la investigación fueron los siguientes: histórico - lógico, en el análisis de la documentación y la literatura sobre la importancia de los estudios y análisis de los estados financieros. El análisis-

síntesis, inducción-deducción y el enfoque sistémico que permitieron diseñar el software para el apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez. Como método empírico, el análisis documental que posibilitó diagnosticar la situación actual que presenta la asignatura Estudio y Análisis de los Estados Financieros. El procesamiento computacional de los resultados, sin excluir el análisis lógico, la analogía, la reflexión y otros procesos mentales que también le son inherentes a toda actividad de investigación científica.

En la investigación se espera como resultado lograr generar los informes necesarios que permita medir la relación causal entre los resultados y efectos económicos, mediante la administración eficiente de los recursos materiales y financieros, lo que trae consigo el incremento de la rentabilidad y la disminución del riesgo de insolvencia.

El trabajo posee una introducción, tres capítulos, conclusiones, recomendaciones, bibliografía y anexos. La introducción expresa los aspectos que justifican la realización del trabajo.

Capítulo I: Fundamentos teóricos y metodológicos sobre la creación de un sistema portable para el apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros para la carrera Contabilidad y Finanzas: En este capítulo se exponen los fundamentos teóricos del presente trabajo. Además, se revisan las tecnologías, lenguajes y herramientas existentes; se describen sus características y el por qué se seleccionaron.

Capítulo II: Descripción del sistema propuesto para contribuir al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez”: Describe todos los elementos que componen el proceso estudiado, utilizando la metodología RUP. Se determinan los actores y trabajadores del negocio, las operaciones principales que producen resultados en forma de casos de uso y su descripción literal, los diagramas de casos de uso y el de modelo de objetos, se determinan los requerimientos funcionales y no funcionales; además, se definen los casos de uso del sistema y se describe cada uno de ellos y los actores del sistema.

Capítulo III: Construcción del sistema propuesto para contribuir al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez”: Se describe el diagrama de clases de diseño, el diagrama de clases persistentes, el modelo de datos y la descripción de cada una de las tablas; se enuncian también los principios de diseño determinando los

estándares usados en la interfaz de la aplicación, la concepción general de la ayuda, el tratamiento de excepciones y cómo es manejada la seguridad; además, se describe la implementación incluyendo el diagrama de despliegue y el diagrama de componentes.

CAPÍTULO I: FUNDAMENTOS TEÓRICOS Y METODOLÓGICOS SOBRE LA CREACIÓN DE UN SISTEMA PORTABLE PARA EL APOYO DOCENTE DE LA ASIGNATURA ESTUDIO Y ANÁLISIS DE LOS ESTADOS FINANCIEROS PARA LA CARRERA CONTABILIDAD Y FINANZAS.

De los gestos a la computadora la humanidad ha empleado medios diversos para la transmisión de la información de un individuo a otro. La contabilidad como ciencia también es parte de la sociedad de la información y en la carrera de Contabilidad y Finanzas en la asignatura Estudio y Análisis de los Estados Financieros, una buena toma de decisiones por parte de los estudiantes conlleva a mejores desempeños como profesionales en el sector empresarial. En el transcurso de este capítulo se realiza un análisis sobre las tendencias metodológicas y/o tecnológicas actuales, se determina el lenguaje de programación a utilizar en la implementación del sistema y se tratan algunos conceptos fundamentales que se usarán durante la investigación.

1.1- Los Estados Financieros

Los Estados Financieros constituyen una representación estructurada de la situación financiera y del rendimiento financiero de la entidad. El objetivo de los Estados Financieros con propósito de información general es suministrar información acerca de la situación financiera, rendimiento financiero y de los Flujos de Efectivo de la entidad, que sea útil a una amplia variedad de usuarios a la hora de tomar sus decisiones económicas (Normas Cubanas de Contabilidad).

Los Estados Financieros de uso interno y externo para las empresas que clasifican en el sector público y privado, son los siguientes:

- El Estado de Situación o Balance General
- Estado de Resultado

Según las Normas Cubanas de Contabilidad el Estado de Situación o Balance General es simplemente un informe sobre la situación financiera de una empresa en un momento determinado donde se indica lo que poseen en bienes y derechos, lo que deben y el monto de su patrimonio, o sea de sus financiamientos propios.

Mientras que el Estado de Resultado muestra la situación económica en un periodo de tiempo determinado; por lo tanto es un documento dinámico. Se basa en registrar todas las partidas de ingreso o de gastos reconocidas en el ejercicio, que se incluirán en el resultado del mismo, al menos que una Norma o una Interpretación establece lo contrario.

1.2- Análisis de las Razones Financieras

El análisis de las razones financieras es una técnica que comenzó a desarrollarse en la primera década del siglo XX, a través de estas se pueden evaluar diversos aspectos del rendimiento y el riesgo de la empresa. El análisis de las razones financieras es una de las formas de medir y evaluar el funcionamiento de la organización y la gestión de sus ejecutivos. Las razones deben compararse dado que en general no son significativas por sí misma. Esta comparación se puede efectuar mediante el uso de:

- Razones pasadas de la misma empresa
- Determinados estándares previamente establecidos
- Razones de otras empresas del mismo organismo.

El análisis de los Estados Financieros normalmente se refiere al cálculo de razones para evaluar el funcionamiento pasado, presente y proyectado de la empresa. El análisis de las razones es la forma más usual de análisis financiero. Los insumos principales para el análisis financiero son el Estado de Resultado, y el Balance General del período o períodos que se consideran. Utilizando los datos que se encuentran en estos estados, pueden calcularse diferentes razones que permitan la evaluación de determinados aspectos del funcionamiento de la empresa. En el [Anexo 1](#) se pueden encontrar todas las razones financieras que se abordan en la investigación.

En la investigación se optó para el análisis comparativo de las razones financieras el método de análisis horizontal, el cual estudia las relaciones entre los elementos financieros para dos juegos de estados, es decir, para los estados financieros de fechas o períodos sucesivos. Por consiguiente, representa una comparación dinámica en el tiempo. El análisis horizontal mediante el cálculo de porcentajes o de razones, se emplea para comparar las cifras de dos estados. Debido a que tanto las cantidades comparadas como el porcentaje o la razón se

presentan en la misma fila o renglón. Este análisis es principalmente útil para revelar tendencias de los Estados Financieros y sus relaciones.

1.3- Sistema económico integrado VERSAT Sarasola

La carrera de Contabilidad y Finanzas en la universidad de Sancti Spíritus “José Martí Pérez” como muchas otras universidades a lo largo del país tiene a la disposición de sus estudiantes el sistema económico VERSAT Sarasola para adiestrarse en su uso, debido a una estrategia del país de migrar la contabilidad general de las empresas a este sistema puramente cubano.

VERSAT Sarasola es un sistema económico integrado que permite a los usuarios la posibilidad de administrar eficientemente los recursos económicos y productivos de que dispone, a partir de la información que en “tiempo real” es procesada a partir de la documentación primaria establecida en cada actividad.

Puede ser utilizada por cualquier tipo de entidad, ya sea esta productiva o presupuestada y además es posible utilizarlo tanto en aquellas que solo posean un solo equipo de cómputo, como las que dispongan de una red con un mayor número de equipos.

Este sistema se encuentra conformado por varios subsistemas como son:

- Subsistema de Configuración.
- Subsistema Costos y Procesos.
- Control de Inventarios.
- Subsistema Finanzas, Caja y Banco.
- Subsistema Control de Activos Fijos.
- Subsistema Facturación.
- Subsistema Nóminas y Salarios.
- Subsistema Planificación y Presupuesto.

- Subsistema Contabilidad General.

El subsistema de Contabilidad General constituye el rector o columna vertebral del sistema, y es además el subsistema más importante para la presente investigación. Las operaciones que se pueden realizar con dicho subsistema son:

- Clasificador de Cuentas, común para el resto de los subsistemas.
- Estructuración de los comprobantes contables.
- Definición de períodos contables y cierre de los mismos.
- Conformación de los comprobantes patrones y recurrentes.
- Asentar todos los comprobantes generados por el resto de los subsistemas, previa revisión de los mismos.
- Conciliación de los saldos de las cuentas que están controladas por otros subsistemas.
- Chequear saldos de las cuentas transitorias.
- Diseño y obtención de los Estados Financieros que se establezcan.
- Ejecución de operaciones automáticas en los cierres de ejercicios.
- Cierre de los períodos y ejercicios contables.
- Ofrece una apreciable cantidad de reportes, incluidos en Balance de Comprobación General.

Al realizar el estudio antes expuesto del sistema VERSAT Sarasola, específicamente del subsistema de Contabilidad General, se pudo constatar que es un sistema muy completo para administrar los recursos económicos de cualquier organización. Pero el subsistema antes explicado, no cuenta con funcionalidades que permitan obtener el Estado Situación (Balance General) y el Estado de Resultado para un período dado, esto dificulta la obtención de las Razones Financieras para un posterior análisis detallado de las mismas. Lo que resulta un

impedimento para que los estudiantes logren agilizar el tiempo de resolución de los ejercicios orientados.

1.4- Metodología utilizada

La metodología de desarrollo de software no es tan solo un proceso, es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software. Es además un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software. En los siguientes subepígrafes se describen las características fundamentales de la metodología utilizada para llevar a cabo el desarrollo del sistema propuesto en la investigación.

1.4.1- Lenguaje Unificado de Modelado

El Lenguaje Unificado de Modelado (*Unified Modeling Language*, UML) es un lenguaje estándar para escribir planos de software, incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables. Puede utilizarse para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software (Booch, 1999)

UML es apropiado para modelar desde sistemas de información en empresas hasta aplicaciones distribuidas basadas en la web, e incluso para sistemas empotrados de tiempo real muy exigentes. Es un lenguaje muy expresivo, que cubre todas las vistas necesarias para desarrollar y luego desplegar tales sistemas. Detrás de cada símbolo en la notación UML hay una semántica bien definida. De esta manera, un desarrollador puede escribir un modelo en UML y otro desarrollador, o incluso otra herramienta, puede interpretar ese modelo sin ambigüedad(Booch, 1999) .

UML es sólo un lenguaje y por tanto es tan sólo una parte de un método de desarrollo de software. Es independiente del proceso, aunque para utilizarlo óptimamente se debería usar en un proceso que fuese dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental tal como el Proceso Unificado de Desarrollo(Booch, 1999).

El vocabulario y las reglas de un lenguaje como UML indican cómo crear y leer modelos bien formados, pero no dicen qué modelos se deben crear ni cuándo se deberían crear. Esta es la

tarea del proceso de desarrollo de software. Un proceso bien definido guiará a sus usuarios al decidir qué artefactos producir, qué actividades y qué personal se emplea para crearlos y gestionarlos, y cómo usar esos artefactos para medir y controlar el proyecto de forma global (Booch, 1999).

1.4.2- Proceso Unificado de Desarrollo

El Proceso Unificado de Desarrollo (*Rational Unified Process* en inglés, habitualmente resumido como RUP) es un proceso de desarrollo de software y junto al UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos (Rumbaugh, 2006)

El RUP se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente, entre otros) y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso) (Rumbaugh, 2006)

El ciclo de vida RUP organiza las tareas en fases e iteraciones. Divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades.

Después del análisis realizado se decidió, utilizar la metodología RUP para llevar a cabo paso a paso el proceso de desarrollo del software propuesto. Esto responde fundamentalmente a que esta metodología se ha convertido en un estándar internacional, al igual que en nuestro país para guiar el proceso de desarrollo de software.

1.5- Tecnologías y lenguajes utilizados

En los siguientes subepígrafes se exponen las tecnologías y lenguajes que se utilizarán para la construcción de un sistema que permita el análisis del Capital de Trabajo, así como las características de los lenguajes de programación elegidos para desarrollar el sistema y el *framework* de persistencia utilizado.

1.5.1- El lenguaje de programación Java

Java es un lenguaje de desarrollo de propósito general, y como tal es válido para realizar todo tipo de aplicaciones profesionales. Incluye una combinación de características que lo hacen único y está siendo adoptado por una multitud de fabricantes como la herramienta básica para el desarrollo de aplicaciones comerciales de gran repercusión (Schildt, 2005).

Una de las características más importantes es que los programas “ejecutables”, creados por el compilador de Java, son independientes de la arquitectura. Se ejecutan indistintamente en una gran variedad de equipos con diferentes microprocesadores y sistemas operativos. Pero además Java se caracteriza por: (Schildt, 2005)

Ser un lenguaje intrínsecamente orientado a objetos.

- Funcionar perfectamente en red.
- Aprovechar características de la mayoría de los lenguajes modernos evitando sus inconvenientes. En particular los del C++.
- Tener una gran funcionalidad gracias a sus bibliotecas (clases).
- No tener punteros manejables por el programador
- El manejo de la memoria no es un problema, la gestiona el propio lenguaje y no el programador.
- Generar aplicaciones con pocos errores posibles.

Muchos fabricantes de software están migrando sus aplicaciones a este nuevo lenguaje. Posiblemente debido a que valoran más las ventajas y, fundamentalmente la posibilidad de desarrollar una sola aplicación para múltiples plataformas así como la mayor facilidad que ofrece Java para el mantenimiento posterior de las aplicaciones. (Schildt, 2005) .

Java es uno de los lenguajes de programación más utilizados en el desarrollo de aplicaciones de código abierto y posee una variada bibliografía para aquellos programadores interesados en el aprendizaje de este lenguaje.

Por todas las razones antes expuestas se selecciona como lenguaje de desarrollo para la implementación del sistema a Java.

1.5.2- El ORM Hibernate

Hibernate es un servicio de consultas y persistencia objeto relacional de alto desempeño muy poderoso. Permite desarrollar clases persistentes siguiendo el paradigma de orientación a objetos, soporta relaciones entre objetos de tipo asociación, herencia simple, polimorfismo, composición, y colecciones. Además expresa consultas en su propia extensión portable de SQL (HQL), o en SQL nativo (Bauer, C., & King, & G, 2005). Hibernate, como todas las herramientas de su tipo busca solucionar el problema de la diferencia entre los dos modelos de datos coexistentes en una aplicación: el usado en la memoria de la computadora (orientación a objetos) y el usado en las bases de datos (modelo relacional). Para lograr esto permite al desarrollador detallar cómo es su modelo de datos, qué relaciones existen y qué forma tienen. Con esta información Hibernate le permite a la aplicación manipular los datos desde la base de datos operando sobre objetos, con todas las características de la programación orientada a objetos.

Hibernate convierte los datos entre los tipos utilizados por Java y los definidos por SQL. También genera las sentencias SQL y libera al desarrollador del manejo manual de los datos que resultan de la ejecución de dichas sentencias, manteniendo la portabilidad entre todos los motores de bases de datos con un ligero incremento en el tiempo de ejecución. (Hibernate, 2009). Entre sus principales características están:

- No intrusivo (estilo POJO)
- Muy buena documentación (fóruns para ayuda, libro)
- Comunidad activa con muchos usuarios
- Transacciones, caché, asociaciones, polimorfismo, herencia, lazy loading, persistencia transitiva, estrategias de fetching.
- Potente lenguaje de Consulta (HQL)
- Fácil testeo

- No es estándar

Por las características antes expuestas el *framework* seleccionado para el desarrollo de la aplicación fue el Hibernate, además de que no se necesita instalar *plugins* para utilizarlo al trabajar con un IDE para desarrollo en lenguaje Java.

1.6- Herramientas de desarrollo utilizadas

En los siguientes subepígrafes se exponen cuáles son las herramientas de desarrollos que se seleccionaron para desarrollar el sistema, se exponen las características principales que propiciaron su elección.

1.6.1- El ambiente de desarrollo de NetBeans

El IDE NetBeans es un proyecto de código abierto de gran éxito con una gran base de usuarios, una comunidad en constante crecimiento, y con cerca de 100 socios en todo el mundo. Sun Microsystems fundó el proyecto de código abierto NetBeans en junio de 2000, aunque en estos momentos pertenece a Oracle.

NetBeans es una base modular y extensible usada como una estructura de integración para crear aplicaciones de escritorio grandes. Empresas independientes asociadas, especializadas en desarrollo de software, proporcionan extensiones adicionales que se integran fácilmente en la plataforma y que pueden también utilizarse para desarrollar sus propias herramientas y soluciones (Netbeans, 2009). La plataforma ofrece servicios comunes a las aplicaciones de escritorio, permitiéndole al desarrollador enfocarse en la lógica específica de su aplicación. Entre las características de la plataforma están:

- Administración de las interfaces de usuario (ej. menús y barras de herramientas)
- Administración de las configuraciones del usuario
- Administración del almacenamiento (guardando y cargando cualquier tipo de dato)
- Administración de ventanas
- *Framework* basado en asistentes (diálogos paso a paso)

El IDE NetBeans es un producto libre y gratuito sin restricciones de uso y de amplias comodidades para el programador.(Netbeans, 2009)Para la realización de este proyecto se utilizó el IDE NetBeans en su versión 6.9, este es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java - pero puede servir para cualquier otro lenguaje de programación.

1.6.2- El sistema de gestión de base de datos SQLite Manager 3.5

SQLITE es un gestor de bases de datos muy ligero y potente. Por sus características se utiliza en una gran variedad de aplicaciones, como Skype, Mozilla Firefox, Adobe Photoshop Elements, el navegador web Opera, ...; y por supuesto en KEME-Contabilidad, donde es una alternativa para el almacenamiento de las contabilidades, junto con MySQL y PostgreSQL.

Las bases de datos bajo SQLITE se almacenan en un archivo que puede ser accedido por un programa monitor interactivo en modo texto denominado "sqlite3". Mediante esta aplicación se pueden efectuar consultas y ediciones utilizando sentencias SQL.

Si estamos acostumbrados a utilizar MySQL o PostgreSQL, sabemos que disponemos de aplicaciones para la administración de estas bases de datos bajo interfaz gráfica. Los máximos exponentes son las aplicaciones PHPmyADMIN y PgAdmin ("Administración gráfica de SQLITE con SQLite Manager,")

1.6.3- El generador de reportes iReport - JasperReport

JasperReports es considerada la mejor herramienta de código libre en Java para generar reportes. Puede entregar ricas presentaciones o diseños en la pantalla, para la impresora o para archivos en formato PDF, HTML, RTF, XLS, CSV y XML.

Está completamente escrita en Java y se puede utilizar en una gran variedad de aplicaciones de Java, incluyendo J2EE o aplicaciones Web, para generar contenido dinámico. JasperReports trabaja en forma similar a un compilador y a un intérprete. El usuario diseña el reporte codificándolo en XML de acuerdo a las etiquetas y atributos definidos en un archivo llamado jasperreports.dtd (parte de JasperReports). Usando XML el usuario define completamente el reporte, describiendo donde colocar texto, imágenes, líneas, rectángulos, cómo adquirir los datos, como realizar ciertos cálculos para mostrar totales, etc.

IReport es un diseñador visual de código libre para JasperReports escrito en Java. Es un programa que ayuda a los usuarios y desarrolladores que usan la librería JasperReports para diseñar reportes visualmente. A través de una interfaz rica y simple de usar, iReport provee las funciones más importantes para crear reportes amenos en poco tiempo. IReport puede ayudar a la gente que no conoce la sintaxis XML para generar reportes.

IReport provee a los usuarios de JasperReports una interfaz visual para construir reportes, generar archivos “jasper” y “print” de prueba. IReport nació como una herramienta de desarrollo, pero puede utilizarse como una herramienta de oficina para adquirir datos almacenados en una base de datos, sin pasar a través de alguna otra aplicación.

1.6.4- El CASE de modelado Visual Paradigm For UML

Visual Paradigm For UML es una Herramienta Case Cruzado de Ciclo de Vida.

Se caracteriza por lo siguiente:

- *Visual Paradigm For UML* es una Herramienta Case que soporta las últimas versiones del mismo, (Lenguaje de Modelado Unificado) y la Notación y Modelado de Procesos de Negocios. Desde un Grupo Administrador de Objetos.
- En adición al soporte de Modelado UML esta herramienta provee el modelado de procesos de negocios, además de un generador de mapeo de objetos-relacionales para los lenguajes de programación Java .NET y PHP.
- Para desarrolladores independientes existe una versión llamada *Community Edition* en la que se caracteriza por ser de uso No Comercial.
- Se integra con las siguientes herramientas Java:
 - Eclipse/IBM WebSphere
 - JBuilder
 - NetBeans IDE
 - Oracle JDeveloper
 - BEA Weblogic

Está disponible en varias ediciones, cada una destinada a unas necesidades: Enterprise, Professional, *Community*, Standard, *Modeler* y Personal

Visual Paradigm para UML se ha actualizado rápidamente en sintonía con el nuevo desarrollo de técnicas de modelado UML 2.1 con el propósito de generar un entorno de modelados visuales en el que se reúnen hoy todas las necesidades tanto de software y tecnología, como de las necesidades de comunicación.

Premiado por Visibilidad en el Entorno de Modelado *Visual Paradigm* para UML es un producto galardonado que facilita a las organizaciones la diagramación visual y el diseño de sus proyectos de sistema mismos que les brinda la posibilidad integrar y desplegar sus aplicaciones empresariales de misión crítica y de sus bases de datos subyacentes.

Esta herramienta ayuda a los equipos de desarrollo de software para sobresalir todo el modelo de acumulación de trabajo así y desplegar el proceso de desarrollo de software, lo que permite maximizar y acelerar tanto las contribuciones individuales como las de equipo

Proporciona el Código y Compatibilidad hasta con 10 lenguajes. *Visual Paradigm* para UML es apoyado por un conjunto de idiomas tanto en la generación del código como en la Ingeniería Inversa por mencionar algunos ejemplos los cuales tiene la capacidad de soporte podríamos hablar de Java, C + +, CORBA IDL, PHP, XML Schema, Ada y Python. Además, apoya la generación del código C #, VB. NET, *Object Definition Language* (ODL), Flash ActionScript, Delphi, Perl, C - Objetivo, y Ruby. Ingeniería Inversa también apoya clase Java. NET .DLL y .exe, JDBC, y archivos de mapeo en Hibernate.(VisualParadigm)

Conclusiones

En este capítulo se realizó una revisión bibliografía para conocer los conceptos fundamentales de la investigación, se exponen las herramientas que se utilizarán para desarrollar el sistema, y de acuerdo a sus características se escogieron las siguientes:

- RUP (Proceso Unificado de Desarrollo) como la metodología más apropiada para el desarrollo del proyecto.
- UML como el lenguaje de modelación utilizado.
- Para la implementación el lenguaje Java, utilizando como herramienta de programación NetBeans IDE 6.9.
- Como *framework* para la persistencia de los datos Hibernate.
- SQLite Manager 3.5 como sistema gestor de bases de datos.
- iReport – JasperReport como *plugin* integrado al IDE NetBeans 6.9 para la generación de reportes.
- *Visual Paradigm For UML* como herramienta de modelado UML.

CAPÍTULO II: DESCRIPCIÓN DEL SISTEMA PROPUESTO PARA CONTRIBUIR AL APOYO DOCENTE DE LA ASIGNATURA ESTUDIO Y ANÁLISIS DE LOS ESTADOS FINANCIEROS DE LA CARRERA CONTABILIDAD Y FINANZAS EN LA UNIVERSIDAD DE SANCTI SPÍRITUS “JOSÉ MARTÍ PÉREZ”

En este capítulo se realiza un estudio del modelo del negocio, este proceso permite una mejor comprensión de la problemática al conocer cuáles son los actores del negocio, los trabajadores, los casos de uso, entre otros. Además se realizan los diagramas de actividades, modelos de objeto, los cuales esclarecen el proceso a automatizar y los pasos que no se deben obviar. También en este capítulo se dejan plasmados los casos de uso del sistema, sus actores, diagramas de casos de uso del sistema así como la descripción de cada uno según lo propuesto por la metodología RUP.

2.1- Modelo del negocio

El modelado de negocio es una técnica para comprender los procesos de negocio de la organización. Además permite identificar los casos de uso y las entidades del negocio relevantes que el software debe soportar, de forma que se puede modelar solo lo necesario para que se comprenda el contexto. (Jacobson, 2000)

El modelo del negocio está soportado por dos tipos de modelos de UML: modelo de casos de uso y modelo de objetos. (Jacobson, 2000)

A continuación se describe el proceso del negocio que se lleva a cabo en la carrera de Contabilidad y Finanzas de la universidad José Martí de Sancti Spíritus, mediante los artefactos propuestos por la metodología RUP y modelados por el lenguaje UML.

2.1.1- Reglas del Negocio

Las reglas del negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto de negocio (Rumbaugh, Booch, & Jacobson, 2006).

Partiendo de lo planteado anteriormente se identificaron las siguientes reglas:

- Un ejercicio está compuesto solamente para dos Estados de Situación y dos Estados de Resultado.
- Cada ejercicio estará evaluado para dos años únicamente.
- Cada indicador o cuenta puede estar repetida solamente dos veces en un mismo ejercicio y cada indicador corresponde a un único año de los dos con que está evaluado el ejercicio.

2.1.2- Modelo de casos de uso del negocio

El modelo de casos de uso del negocio es el encargado de describir los procesos de una empresa utilizando los casos de uso y los actores, en correspondencia con los procesos del negocio y los clientes (Jacobson, 2000). Esto se define a través del diagrama de casos de uso y el diagrama de actividades.

2.1.3- Actores del negocio

Se considera actor del negocio a cualquier individuo, grupo, entidad, organización, máquina o sistema de información externo; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados (Jacobson, 2000).

Tabla 1: Actores del negocio

Actores del Negocio	Descripción
Alumno	Es el responsable de calcular el ejercicio

2.1.4- Diagrama de casos de uso del negocio

Los diagramas de casos de uso se utilizan para especificar las funcionalidades y el comportamiento de un sistema mediante su interacción con los usuarios y otros sistemas. O sea es un diagrama que muestra la relación entre los actores y los casos de uso en un sistema (Jacobson, 2000).

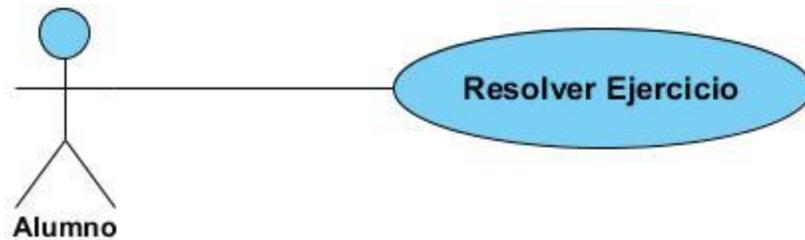


Figura 1 : Diagrama de casos de uso del negocio

2.1.5- Trabajadores del negocio

Un trabajador es una abstracción de una persona o grupo de personas, una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio, manipulando entidades del mismo y representando un rol (González, 2005)

Los trabajadores identificados en el negocio son los siguientes:

Tabla 2: Trabajadores del negocio

Trabajadores del Negocio	Descripción
Profesor	Es quien se encarga de orientar los Ejercicios

2.1.6- Diagramas de actividades

Un diagrama de actividades es un diagrama que muestra el flujo de actividad a actividad que trata la vista dinámica de un sistema. Es un caso especial de diagrama de estados en el cual todos o casi todos los estados son estados de acción y en el cual todas o casi todas las transiciones son disparadas por la terminación de las acciones en los estados origen (Rumbaugh, 2006). El diagrama de actividades puede observarse en el [Anexo 2](#).

2.1.7- Modelo de objetos del negocio

El modelo de objetos del negocio se utiliza para describir la participación de los trabajadores y entidades del negocio, y su colaboración en la realización del negocio. Un modelo de objetos

del negocio es un modelo interno a un negocio. Describe como cada caso de uso del negocio es llevado a cabo por parte de un conjunto de trabajadores que utilizan un conjunto de entidades del negocio y unidades de trabajo (González, 2005). Una entidad del negocio es algo que los trabajadores toman, inspeccionan, manipulan o producen en un caso de uso del negocio (González, 2005). La siguiente figura muestra el modelo de objetos del negocio.



Figura 2 : Modelo de Objetos del Negocio

2.2- Requerimientos funcionales

(González, 2005). Plantea que un requerimiento funcional expresa una especificación más detallada de las responsabilidades del sistema que se propone. Ellos permiten determinar, de una manera clara lo que debe hacer el sistema, siempre basándose en las necesidades de los usuarios. Los requerimientos funcionales del sistema propuesto son:

RF-1: Mostrar Reporte de las Razones de Liquidez.

- 1.1- Mostrar Liquidez Inmediata.
- 1.2- Mostrar Liquidez a Corto Plazo o Prueba Acida.
- 1.3- Mostrar Liquidez de Tesorería.
- 1.4- Mostrar Liquidez Media.
- 1.5- Mostrar Liquidez total o General.
- 1.6- Mostrar Total del Capital de Trabajo.
- 1.7- Mostrar Disponibilidad.

RF-2: Mostrar Reporte de las Razones de Solvencia.

- 2.1- Mostrar Solvencia a Largo Plazo.
- 2.2- Mostrar Solvencia Total.

RF-3: Mostrar Reporte de las Razones de Solidez.

- 3.1- Mostrar Ratios de Solidez.
- 3.2- Mostrar Estabilidad de los Activos Fijos.
- 3.3- Mostrar Estabilidad Propia de los Activos Fijos.

RF-4: Mostar Reporte de las Razones de Endeudamiento.

- 4.1- Mostar Endeudamiento.
- 4.2- Mostrar Endeudamiento Total.
- 4.3- Mostrar Grado de Endeudamiento.
- 4.4- Mostrar Capacidad para atender el servicio de deudas.
- 4.5- Mostrar Costo de la Deuda.
- 4.6- Mostar la Calidad de la Deuda.

RF-5: Mostar Reporte de las Razones de Rotación.

- 5.1- Mostrar Índice de Rotación de la Empresa.
- 5.2- Mostrar Índice de Rotación de los Activos Fijos.
- 5.3- Mostrar Índice de Rotación del Activo Circulante.
- 5.4- Mostrar Índice de Rotación del Capital de Trabajo.
- 5.5- Mostrar Índice de Rotación del Ciclo de Cobros.
- 5.6- Mostrar Índice de Rotación del Ciclo de Pagos.

RF-6: Mostrar Reportes de las Razones de Rentabilidad.

- 6.1- Mostrar Rentabilidad de los Ingresos o del Margen Neto.
- 6.2- Mostrar Rentabilidad Económica.
- 6.3- Mostrar Rentabilidad Económica Neta.
- 6.4- Mostrar Rentabilidad Financiera.
- 6.5- Mostrar la Composición del Índice de Rentabilidad Económica.
- 6.6- Mostrar la Composición del Índice de Rentabilidad Financiera.
- 6.7- Mostar Ratios de Apalancamiento Financiero.

RF-7: Mostrar Reporte sobre Ventas.

- 7.1- Mostrar la Productividad.

RF-8: Gestionar Activos Circulantes.

- 8.1- Insertar Activos Circulantes.
- 8.2- Modificar Activos Circulantes.
- 8.3- Eliminar Activos Circulantes

RF-9: Gestionar Activos A Largo Plazo.

- 9.1- Insertar Activos A Largo Plazo.
- 9.2- Modificar Activos A Largo Plazo.
- 9.3- Eliminar Activos A Largo Plazo.

RF-10: Gestionar Activos Fijos.

- 10.1- Insertar Activos Fijos.
- 10.2- Modificar Activos Fijos.
- 10.3- Eliminar Activos Fijos.

RF-11: Gestionar Activos Diferidos.

- 11.1- Insertar Activos Diferidos.
- 11.2- Modificar Activos Diferidos.
- 11.3- Eliminar Activos Diferidos.

RF-12: Gestionar Otros Activos.

- 12.1- Insertar Otros Activos.
- 12.2- Modificar Otros Activos.
- 12.3- Eliminar Otros Activos.

RF-13: Gestionar Cuentas Reguladoras de Activos.

- 13.1- Insertar Cuentas Reguladoras de Activos.
- 13.2- Modificar Cuentas Reguladoras de Activos.
- 13.3- Eliminar Cuentas Reguladoras de Activos.

RF-14: Gestionar Pasivos Circulantes.

- 14.1- Insertar Pasivos Circulantes.
- 14.2- Modificar Pasivos Circulantes.
- 14.3- Eliminar Pasivos Circulantes.

RF-15: Gestionar Pasivos A Largo Plazo.

- 15.1- Insertar Pasivos A Largo Plazo.
- 15.2- Modificar Pasivos A Largo Plazo.
- 15.3- Eliminar Pasivos A Largo Plazo.

RF-16: Gestionar Pasivos Diferidos.

- 16.1- Insertar Pasivos Diferidos.
- 16.2- Modificar Pasivos Diferidos.
- 16.3- Eliminar Pasivos Diferidos.

RF-17: Gestionar Otros Pasivos.

- 17.1- Insertar Otros Pasivos.
- 17.2- Modificar Otros Pasivos.
- 17.3- Eliminar Otros Pasivos.

RF-18: Gestionar Patrimonio Neto o Capital Contable.

- 18.1- Insertar Patrimonio Neto o Capital Contable.
- 18.2- Modificar Patrimonio Neto o Capital Contable.
- 18.3- Eliminar Patrimonio Neto o Capital Contable.

RF-19: Gestionar Gastos de Producción.

- 19.1- Insertar Gastos de Producción.
- 19.2- Modificar Gastos de Producción.
- 19.3- Eliminar Gastos de Producción.

RF-20: Gestionar Cuentas Nominales Deudoras (Excepto Empresas de Seguro).

- 20.1- Insertar Cuentas Nominales Deudoras.
- 20.2- Modificar Cuentas Nominales Deudoras.
- 20.3- Eliminar Cuentas Nominales Deudoras.

RF-21: Gestionar Cuentas Nominales Acreedoras (Excepto Empresas de Seguro).

- 21.1- Insertar Cuentas Nominales Acreedoras.
- 21.2- Modificar Cuentas Nominales Acreedoras.

21.3- Eliminar Cuentas Nominales Acreedoras.

RF-22: Gestionar Cuentas Nominales Deudoras (Empresas de Seguro).

22.1- Insertar Cuentas Nominales Deudoras.

22.2- Modificar Cuentas Nominales Deudoras.

22.3- Eliminar Cuentas Nominales Deudoras.

RF-23: Gestionar Cuentas Nominales Acreedoras (Empresas de Seguro).

23.1- Insertar Cuentas Nominales Acreedoras.

23.2- Modificar Cuentas Nominales Acreedoras.

23.3- Eliminar Cuentas Nominales Acreedoras.

RF-24: Gestionar Cuentas de Compras.

24.1- Insertar Cuentas de Compras.

24.2- Modificar Cuentas de Compras.

24.3- Eliminar Cuentas de Compras.

2.3- Requerimientos no funcionales

Los requisitos no funcionales especifican propiedad del sistema, como restricciones del entorno o de implementación, rendimiento, dependencias de la plataforma, mantenibilidad, extensibilidad o fiabilidad (Rumbaugh, 2006).

Requerimientos de apariencia o interfaz externa

La interfaz estará diseñada de modo tal que el usuario pueda tener en todo momento el control de la aplicación lo que le permitirá ir de un punto a otro dentro del sistema con gran facilidad. Se cuidará porque la aplicación sea lo más interactiva posible.

Requerimientos de usabilidad

El sistema podrá ser usado por cualquier persona, la cual tendrá acceso a todas las funcionalidades de la aplicación sin ser requerido ningún tipo de autenticación.

Requerimientos de rendimiento

La eficiencia del sistema estará determinada por el aprovechamiento de la velocidad de las consultas en la base de datos por el ORM Hibernate. El sistema deberá ser lo más estable y confiable posible.

Requerimientos de soporte

Se requiere que el producto reciba mantenimiento ante cualquier fallo que ocurra. No será necesario instalar la aplicación. El sistema deberá dar las posibilidades a futuras mejoras y nuevas opciones que se le quieran incorporar.

Requerimientos de portabilidad

La plataforma seleccionada para desarrollar la aplicación fue Windows. Sin embargo el cliente podrá ser ejecuto en cualquiera de las familias de los sistemas operativos Linux que soporten el lenguaje Java.

Confiabilidad

El sistema en casos de fallos debe garantizar que las pérdidas de información sean mínimas y debe brindar garantía de un tratamiento adecuado de las excepciones y validación de las entradas del usuario.

Requerimientos de ayuda y documentación en línea

El sistema contará con una ayuda que explicará de manera clara y sencilla al usuario todas las funcionalidades del sistema.

La ayuda quedará conformada por un menú general que le facilitará al usuario poder ir de un lugar a otro sin perderse.

La ayuda se realizará mediante páginas Web, las cuales se cargarán en un mini navegador del propio sistema, esto trae como beneficio que no se ponga en peligro la portabilidad.

Requerimientos de Software

Se debe disponer de un sistema operativo compatible. El sistema podrá ejecutarse en entornos Windows y Linux siempre y cuando tengan instalada la Máquina Virtual de Java (JVM).

Requerimientos de Hardware

Para la puesta en práctica del proyecto se requieren máquinas con los siguientes requisitos:

Las computadoras usadas por los usuarios requerirán como mínimo:

Un Procesador Pentium III.

128 Mb de Memoria RAM.

Restricciones en el diseño y la implementación

Se utilizarán herramientas de desarrollo que garanticen la calidad de todo el ciclo de desarrollo del producto.

2.4- Modelo de casos de uso del sistema

El modelado de casos de uso es la técnica más simple que emplean los desarrolladores de software para modelar los requisitos del sistema desde la perspectiva del usuario. El modelo de casos de uso consiste en actores y casos de uso. Los actores representan usuarios y otros subsistemas que interactúan con el sistema y los casos de uso representan el comportamiento del mismo, los escenarios que el sistema atraviesa en respuesta a un estímulo desde un actor. ("Popkin Software and System," 2005)

2.4.1- Actores del sistema

Un actor no es más que un rol que juega un usuario de caso de uso cuando interactúan con estos casos de uso. Los actores representan a terceros fuera del sistema que colaboran con el mismo. Una vez que se han identificado los actores del sistema, se ha identificado el entorno externo del sistema. (Rumbaugh, 2006)

Teniendo en cuenta lo planteado anteriormente se definieron los siguientes actores del sistema:

Tabla 3: Actores del sistema

Actores del sistema	Descripción
Usuario	Interesado en generar los reportes vinculados a los Estados Financieros.

2.4.2- Casos de uso del sistema.

Cada forma en que los actores usan el sistema se representa con un caso de uso. Los casos de uso son “fragmentos” de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. Especifican una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia (Jacobson, 2000). Para este sistema se definieron los siguientes casos de uso:

- 1- Mostrar Reporte de las Razones de Liquidez.
- 2- Mostrar Reporte de las Razones de Solvencia.
- 3- Mostrar Reporte de la Razones de Solidez.
- 4- Mostar Reporte de las Razones de Endeudamiento.
- 5- Mostar Reporte de la Razones de Rotación.
- 6- Mostrar Reportes de las Razones de Rentabilidad.
- 7- Mostrar Reporte sobre Ventas.
- 8- Gestionar Activos Circulantes.
- 9- Gestionar Activos A Largo Plazo.
- 10- Gestionar Activos Fijos.
- 11- Gestionar Activos Diferidos.
- 12- Gestionar Otros Activos.
- 13- Gestionar Cuentas Reguladoras de Activos.
- 14- Gestionar Pasivos Circulantes.
- 15- Gestionar Pasivos A Largo Plazo.
- 16- Gestionar Pasivos Diferidos.
- 17- Gestionar Otros Pasivos.
- 18- Gestionar Patrimonio Neto o Capital Contable.
- 19- Gestionar Gastos de Producción
- 20- Gestionar Cuentas Nominales Deudoras (Excepto Empresas de Seguro).
- 21- Gestionar Cuentas Nominales Acreedoras (Excepto Empresas de Seguro).
- 22- Gestionar Cuentas Nominales Deudoras (Empresas de Seguro).
- 23- Gestionar Cuentas Nominales Acreedoras (Empresas de Seguro).
- 24- Gestionar Cuentas de Compras.

2.4.3- Diagramas de casos de uso del sistema.

Para facilitar el trabajo con los casos de uso y la organización de los elementos se agrupan los casos de uso en paquetes, los cuales se muestran a continuación.

Tabla 4: Descripción de los paquetes

Nombre del paquete	Funcionalidades	Criterio de agrupamiento	Actores
Operaciones	<ul style="list-style-type: none"> • Gestionar Activos Circulantes. • Gestionar Activos A Largo Plazo. • Gestionar Activos Fijos. • Gestionar Activos Diferidos. • Gestionar Otros Activos. • Gestionar Cuentas Reguladoras de Activos. • Gestionar Pasivos Circulantes. • Gestionar Pasivos A Largo Plazo. • Gestionar Pasivos Diferidos. • Gestionar Otros Pasivos. • Gestionar Patrimonio Neto o Capital Contable. • Gestionar Gastos de Producción • Gestionar Cuentas Nominales Deudoras (Excepto Empresas de Seguro). • Gestionar Cuentas Nominales Acreedoras (Excepto Empresas de Seguro). • Gestionar Cuentas Nominales Deudoras (Empresas de Seguro). • Gestionar Cuentas Nominales 	Funcionalidades (Requerimientos funcionales)	Alumno

	<p>Acreedoras (Empresas de Seguro).</p> <ul style="list-style-type: none"> • Gestionar Cuentas de Compras. 		
Reportes	<ul style="list-style-type: none"> • Mostrar Reporte de las Razones de Liquidez. • Mostrar Reporte de las Razones de Solvencia. • Mostrar Reporte de la Razones de Solidez. • Mostar Reporte de las Razones de Endeudamiento. • Mostar Reporte de la Razones de Rotación. • Mostrar Reportes de las Razones de Rentabilidad. • Mostrar Reporte sobre Ventas. 	<p>Funcionalidades (Requerimientos funcionales)</p>	Alumno

En la siguiente figura se muestran los paquetes que agrupan a los casos de uso.



Figura 3 : Casos de uso por paquetes.

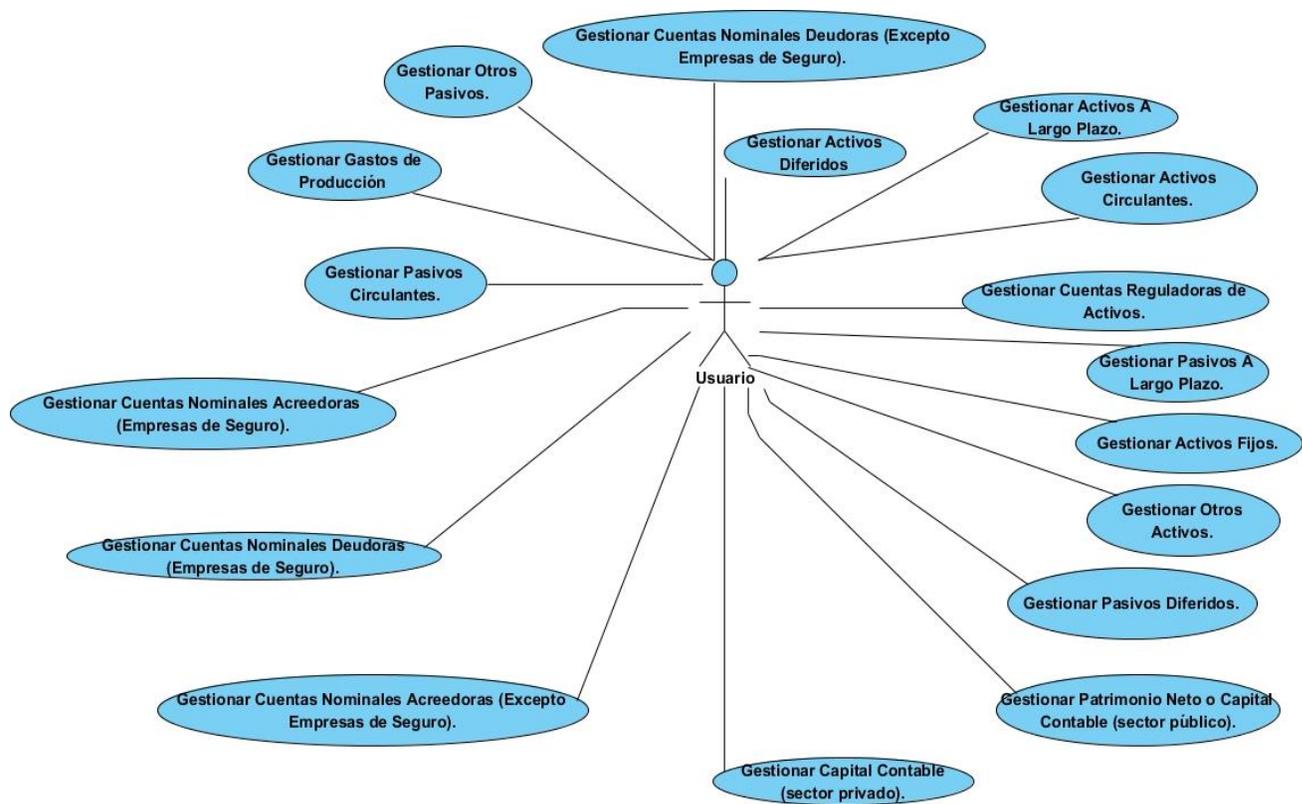


Figura 4 : Diagrama casos de uso del Sistema Paquete Operaciones.



Figura 5: Diagrama casos de uso del Sistema Paquete Reportes.

La descripción de los casos de uso del sistema se encuentran en el [Anexo 3](#).

Conclusiones

En este capítulo se deja claro cómo funciona el negocio a través de los artefactos proporcionados por la metodología RUP. Además contiene una descripción general del sistema identificando los requerimientos funcionales y no funcionales y los casos de uso. La construcción de todos estos artefactos propició que:

- Se esclareciera cómo es el flujo de eventos que se realiza en cada uno de los casos de uso.
- Se establecieron las relaciones de cada uno de los actores del sistema con los casos de uso.

CAPÍTULO III: CONSTRUCCIÓN DEL SISTEMA PROPUESTO PARA CONTRIBUIR AL APOYO DOCENTE DE LA ASIGNATURA ESTUDIO Y ANÁLISIS DE LOS ESTADOS FINANCIEROS DE LA CARRERA CONTABILIDAD Y FINANZAS EN LA UNIVERSIDAD DE SANCTI SPÍRITUS “JOSÉ MARTÍ PÉREZ”

La parte del proceso de desarrollo de software cuyo principal propósito es decidir cómo se llevará a cabo el sistema, es el diseño. Durante esta etapa se toman decisiones importantes que conllevan al efectivo cumplimiento de los requerimientos funcionales y la obtención de un software con calidad. En el presente capítulo se plasman los resultados de la etapa del diseño. Se presentan diagramas como por ejemplo: modelo conceptual y físico de la base de datos, diagrama de entidades de clases a partir de la base de datos, diagrama de componente y de despliegue. Además se describen los principios de diseño aplicados.

3.1- Diseño de clases persistentes

En el diagrama de clases persistentes aparecen las clases que persisten, las cuales poseen la capacidad de mantener su valor en el espacio y en el tiempo(Rumbaugh, 2006). Está compuesto por clases, asociaciones y atributos; interfaces, con sus operaciones y constantes; métodos; información sobre los tipos de atributos, entre otros. Este diagrama describe las clases entidades las cuales son interpretadas por el *framework hibernate* creado el modelo de datos para la base de datos, permitiendo trabajar con bases de datos orientadas a objetos. El diagrama de clases persistentes puede ser consultado en el [Anexo 4](#).

3.2- Modelo de Datos

El modelo físico de datos, representa la estructura o descripción física de las tablas de la base de datos, obtenido a partir del modelo lógico de datos. (Rumbaugh, Booch, & Jacobson, 2006)

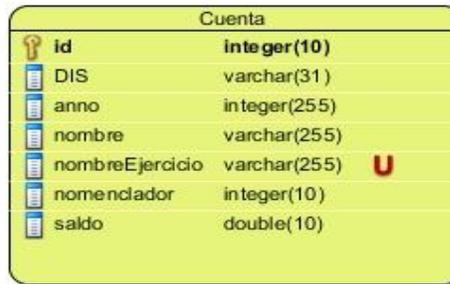


Figura 6: Modelo de Datos

3.3- Diseño de clases del Sistema

Una clase de diseño es una abstracción de una clase o construcción similar en la implementación del sistema. (Rumbaugh, 2006).

Un diagrama de clases es una colección de elementos declaratorios del modelo, como clases, tipos y sus relaciones; conectados unos a otros y a sus contenidos en forma de grafo (Rumbaugh, 2006).

El diagrama de clases de diseño describe gráficamente las especificaciones de las clases de software y de las interfaces en una aplicación. A diferencia del modelo conceptual, un diagrama de este tipo contiene las definiciones de las entidades del software en vez de conceptos del mundo real. Los Diagramas de clases del Diseño se pueden encontrar en el [Anexo 5](#).

3.4- Principios de Diseño

El diseño de la interfaz de una aplicación, el formato de los reportes, la concepción de la ayuda, el tratamiento de errores y la seguridad, tienen gran influencia en el éxito o fracaso de una aplicación. A continuación se describen los principios de diseño seguidos para el desarrollo del sistema en cuestión.

3.4.1- Diseño de la interfaz del sistema

Uno de los aspectos más relevantes en el uso de un sistema que lo puede llevar a un éxito o a un fracaso es la consistencia de la interfaz de usuario.

La interfaz de SistEA (Sistema para el apoyo a la docencia de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas de la Universidad de Sancti Spiritus “José Martí Pérez”) se concibe con la mayor sencillez posible, de manera que el trabajo

con el sistema sea fácil y ameno. Se utilizan colores agradables a la vista y una letra legible para su rápida comprensión, además se utiliza un lenguaje conocido por el usuario, sin emplear términos informáticos.

El software brinda un menú superior en la ventana principal que brindará las diferentes funciones que puede realizar el usuario. Además consta con una ayuda que aparece de manera permanente en todas las ventanas en la cual se encuentran descritas todas las funcionalidades del sistema.

3.4.2- Tratamiento de errores

El sistema propuesto valida constantemente la información, con el propósito de minimizar las posibilidades de introducir errores por parte del usuario. En caso de errores se le comunica al usuario a través de un mensaje de alerta en un lenguaje fácil de comprender.

3.4.3- Concepción general de la ayuda

El sistema cuenta con una ayuda capaz de explicar al usuario cómo es la manipulación y funcionamiento del software. La ayuda está formada por un menú interno que contiene todas las funcionalidades del sistema. El usuario podrá navegar por cada uno de y consultarlo en cualquier momento ya que se encuentra ubicada en el menú principal de la aplicación.

3.4.4- Estándares de codificación

Las convenciones o estándares de codificación son pautas de programación que no están enfocadas a la lógica del programa, sino a su estructura y apariencia física para facilitar la lectura, comprensión y mantenimiento del código. (Humphrey, 2001)

Actualmente se hallan estándares de codificación para la mayoría de los lenguajes existentes, su uso, partiendo de las convenciones definidas, permite una mejor comunicación entre los programadores creando las condiciones para la reusabilidad y el mantenimiento de los sistemas. Por lo que se decide que las variables, nombres de funciones, de consultas y objetos del documento son cortos, claros y describen su propósito. Los nombres de las clases se escriben con mayúscula, las variables con minúsculas y las funciones que están compuestas por más de una palabra se escribe primero con minúscula y la primera letra de las demás palabras con mayúscula. Los objetos o tipos de control se nombran según el valor de su contenido. Los inicios y cierre de ámbito se encuentran alineados debajo de la declaración a la

que pertenecen y se evitan si hay sólo una instrucción. Los signos lógicos y de operación se separan por un espacio antes y después de los mismos.

3.5- Modelo de implementación

El modelo de implementación describe cómo los elementos del modelo del diseño, se implementan en términos de componentes, describe cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en los lenguajes de programación utilizados y cómo dependen unos componentes de otros (Jacobson, 2000).

3.5.1- Diagrama de Despliegue

El diagrama de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuyen las funcionalidades entre los nodos de cómputo (Jacobson, 2000). El modelo representa un nodo donde se visualiza la aplicación en el lado del cliente.

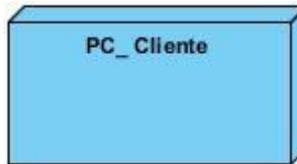


Figura 7: Diagrama de despliegue

3.5.2- Diagrama de Componentes

Un diagrama de componentes muestra un conjunto de elementos del modelo, se utiliza para modelar la vista estática de un sistema, muestra la organización y dependencias lógicas entre los componentes del software ya sean bibliotecas, ejecutables o componentes binarios. Proporciona un alto nivel de comprensión de la arquitectura del sistema que será construido, permitiendo a los desarrolladores formalizar un plan de trabajo para la ejecución, y tomar decisiones sobre la asignación de tareas y habilidades necesarias para las mejoras futuras del software.

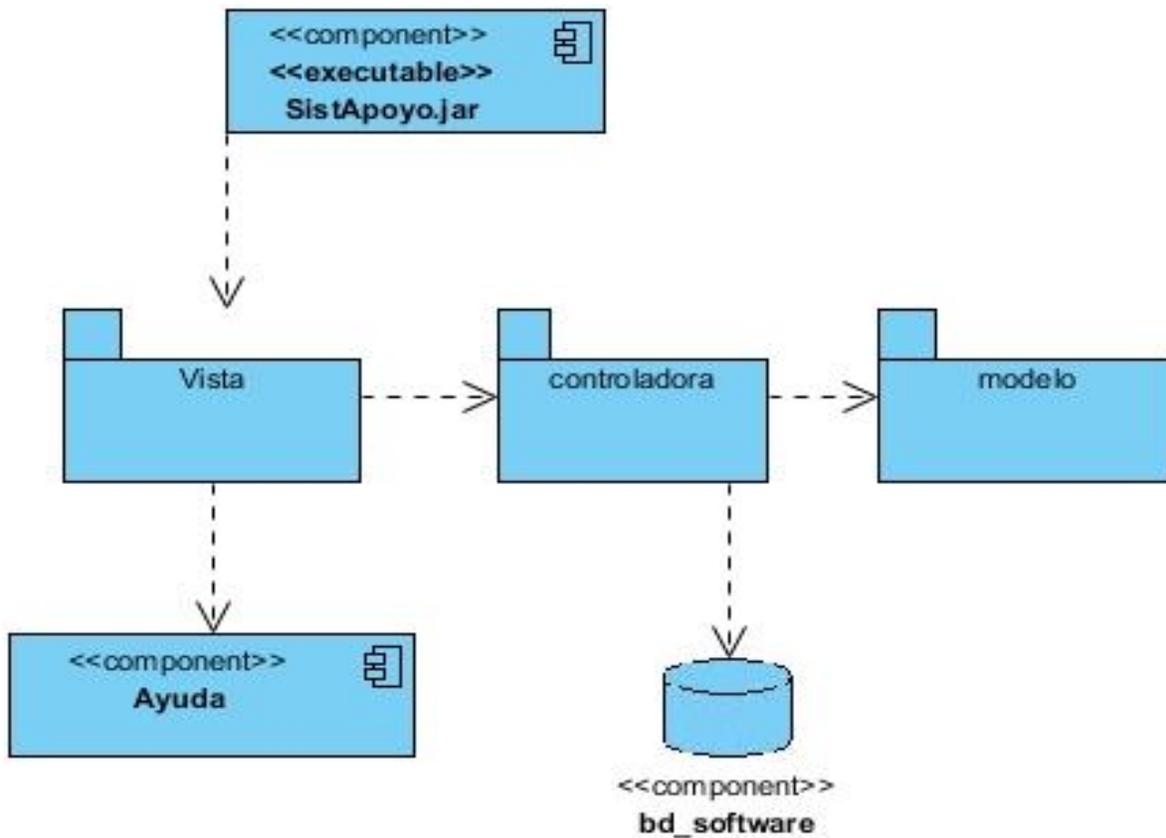


Figura 8: Diagrama de componentes

Tabla 5: Descripción de los componentes SistApoyo.jar

Componente	Descripción
SistApoyo.jar	Este componente representa el ejecutable del sistema.
Vista	Dentro de este subsistema se encuentran las clases que generan las ventanas para cargar o crear nuevo ejercicio, gestionar las cuentas contables y la generación de reportes.
Controladora	Este subsistema representa los controladores para actualizar las vistas y gestionar las clases entidades.
Modelo	Este subsistema representa la lógica del negocio, dentro de él se encuentran las clases entidades con las entidades del

	negocio y sus atributos.
db_software	Representa la base de datos del sistema.
Ayuda	Este componente representa la ayuda del sistema.

3.6- Pruebas

Las pruebas son un flujo de trabajo donde se verifica el resultado de la implementación probando cada construcción, incluyendo tanto construcciones internas como intermedias, así como las versiones finales del sistema a ser entregadas a terceros (Jacobson, 2000). En los siguientes subepígrafos se exponen las planificaciones de prueba, los procedimientos de prueba y los casos de prueba para algunos de los casos de uso más importantes del sistema según el juicio del autor.

3.6.1- Planificación de prueba

El objetivo de la planificación de las pruebas es la representación de una estrategia de prueba, planificar su esfuerzo y estimar los requisitos de la misma. Se organiza el plan de pruebas utilizando un rango de valores de entrada. El modelo de casos de uso y los requisitos que debe tener el sistema para ejecutar correctamente el proceso, los mismos ayudan a definir un tipo adecuado de pruebas y a estimar el esfuerzo necesario para llevar a cabo las pruebas. Se desarrolla una estrategia de prueba para la iteración donde se decide qué tipo de pruebas ejecutar, cómo y cuándo ejecutarlas y cómo determinar si el esfuerzo de prueba tiene éxito (Jacobson, 2000)

A continuación se exponen la planificación de pruebas para dos de los casos de uso del sistema.

Caso de uso: Mostrar Razones de Rotación.

Casos de prueba:

Insertar los días por los cuales se deben multiplicar los ratios ciclo de cobro y ciclo de pago.

Caso de uso: Gestionar Activos Circulantes.

Casos de prueba: Insertar las cuentas que pertenecen a Activos Circulantes.

3.6.2- Procedimientos de prueba

Los procedimientos de pruebas especifican como realizar uno o varios casos de pruebas. Estos incluyen información adicional, como los valores de entrada del caso de uso a utilizar, la forma en la que estos valores han de ser introducidos en el interfaz de usuario y lo que hay que verificar. A continuación se exponen los procedimientos de pruebas para los casos de uso del sistema seleccionados durante la planificación de pruebas.

Procedimiento de prueba del caso de uso: Mostar Razones de Rotación.

- 1) Hacer clic en el ítem Razones de Rotación que se encuentra dentro de la barra de menú en el menú Reportes.
- 2) El sistema muestra una caja de texto para que el usuario inserte los días por los cuales se deberá multiplicar
- 3) El Reporte se llena con los datos insertados.
- 4) El sistema muestra el reporte en el cual se observan los resultados de los ratios que pertenecen a las Razones de Rotación para que con estos el usuario pueda proceder a analizarlos.

Procedimiento de prueba del caso de uso: Gestionar Activos Circulantes.

- 1) Hacer clic en el ítem Activos Circulantes que se encuentra en el menú Activos dentro de la barra de menú en el menú Cuentas.
- 2) El sistema muestra una ventana llamada Gestionar Activos Circulantes.
- 3) Se muestra la ventana Gestionar Activos Circulantes. La ventana muestra una tabla donde se podrán visualizar todas las cuentas entradas al sistema así como un combobox donde el usuario podrá entrar el nombre de la cuenta, además se muestran los campos donde el usuario deberá insertar el saldo, el nomenclador y el año de la cuenta así como los botones Nuevo, Insertar, Actualizar y Eliminar.
- 4) El usuario deberá dar clic en el botón nuevo para comenzar a insertar una cuenta.
- 5) Después de presionar el botón Nuevo los campos que contiene la ventana se desbloquearán por lo que el usuario deberá escoger un nombre del combobox, insertar el saldo, el nomenclador y el año de la cuenta.
- 6) Presionar el botón insertar el sistema insertará los datos de la cuenta en una lista de Cuentas el usuario podrá ver un mensaje con “cuenta adicionada con éxito”.
- 7) Se podrá visualizar en la tabla lo realizado.

3.6.3- Casos de prueba

Los casos de pruebas especifican la forma de probar el sistema, incluyendo la entrada o resultado con la que se ha de probar y las condiciones bajo las que ha de probarse. Es un conjunto de entradas y resultados esperados que ejercitan a un componente con el propósito de causar fallas y detectar defectos. A continuación se exponen los casos de pruebas realizados a los diferentes casos de uso del sistema seleccionados para la planificación y los procedimientos de prueba.

Tabla 6: Caso de prueba para el caso de uso Mostrar Razones de Rotación.

Caso de uso: Mostrar Razones de Rotación.
Caso de prueba: Insertar los días por los cuales se deben multiplicar los ratios ciclo de cobro y ciclo de pago.
Entrada: El usuario intenta presionar el botón ok habiendo insertado letras en la caja de texto donde se le pide insertar los días
Resultado: Se le muestra al usuario un mensaje diciéndole que solo se pueden insertar números.
Condiciones: Existir en la base de datos los datos necesarios para calcular los ratios.

Tabla 7: Caso de prueba para el caso de uso Gestionar Activos Circulantes.

Caso de uso: Gestionar Activos Circulantes.
Caso de prueba: Insertar las cuentas que pertenecen a Activos Circulantes.
Entrada: El usuario intenta presionar el botón Insertar sin haber llenado los campos necesarios para las cuentas.
Resultado: Se le muestra al usuario un mensaje diciéndole que debe rellenar los campos.
Condiciones: Deben estar insertados correctamente todos los campos necesarios para insertar una cuenta.

CONCLUSIONES

Durante el desarrollo de este capítulo se elaboró el diagrama de componente en el cual se graficó de manera clara las relaciones entre los subsistemas, los componentes y la base de datos; el diagrama de despliegue y los modelos lógicos y físicos de la base de datos. Además se describieron los principios de diseño seguidos, específicamente el diseño de la interfaz de usuario, los estándares de codificación, la concepción de la ayuda, el tratamiento de excepciones y la seguridad del sistema.

CONCLUSIONES

1. Se realizó un estudio teórico metodológico con el cual se recopilaron los principales referentes para realizar este producto lo que permitió determinar una metodología adecuada, así como las herramientas y tecnologías a utilizar en el desarrollo del sistema para contribuir al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez”.
2. Se diseñó un sistema informático para contribuir al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez” sustentado en la programación orientada a objeto, teniendo en cuenta los requerimientos del negocio y las necesidades del cliente.
3. Se implementó un sistema informático para contribuir al apoyo docente de la asignatura Estudio y Análisis de los Estados Financieros de la carrera Contabilidad y Finanzas en la Universidad de Sancti Spíritus “José Martí Pérez”, el cual posibilita aprovechar el tiempo en las clases prácticas.

RECOMENDACIONES

- ✚ Tener cargados los datos de los ejercicios previamente en un fichero para así agilizar el proceso de entrada al sistema y con esto tener un mayor aprovechamiento del tiempo de las clases.
- ✚ Diseñar e Implementar reportes que permitan mostrar en gráficos estadísticos los comportamientos de la varianza para cada ejercicio resuelto.

BIBLIOGRAFÍA

- Administración gráfica de SQLITE con SQLite Manager. from <http://www.RibosaBlog.com>
- Bauer, C., & King, & G. (2005). *Hibernate in Action*: Manning Publications Co.
- Booch, G., Rumbaugh, J & Jacobson, I. (1999). *El Lenguaje Unificado de Modelado*. Boston, MA: Addison Wesley.
- González, A. (2005). *Modelamiento del negocio*. Centro de Estudios de Ingeniería de Sistemas (CEIS).
- Hibernate. (2009). Recuperado el 14 de enero de 2013, from <http://www.hibernate.org>
- Humphrey, W. S. (2001). *Introducción al Proceso de Software Personal*. Madrid: Pearson Educación, S.A.
- Jacobson, I., Booch,G...,& Rumbaugh,J. (2000). *El proceso unificado de desarrollo*. Madrid: Addison Wesley.
- Netbeans. (2009). Recuperado el 14 de mayo de 2011, from <http://www.netbeans.org>
- Popkin Software and System. (2005). Recuperado el 10 de mayo de 2011, de Modelado de sistemas con UML
- from <http://es.tldp.org/tutoriales/doc-modelado-sistemas-UML-multiple-html/c124.html>
- Rumbaugh, J., Booch, G., & Jacobson, I. (2006). *El proceso unificado de desarrollo de software*. La Habana: Félix Varela.
- Schildt, H. (2005). *Java™:A Beginner's Guide*. New York: McGraw-Hill Co.
- VisualParadigm. Visual Paradigm For Uml. from <http://wwwVisualParadigmForUml.htm>

ANEXOS

Anexo 1: Despliegue de los indicadores de las Razones Financieras.

Anexo 1.1: Despliegue de los indicadores de las Razones de Liquidez

Clasificación	Cálculo
Liquidez general o Solvencia	Activo Circulante/Pasivo Circulante
Liquidez inmediata o Prueba Ácida	$(\text{Activo Circulante} - \text{Inventario}) / \text{Pasivo Circulante}$
Liquidez Disponible o Tesorería	Efectivo/Pasivo Circulante

Anexo 1.2: Despliegue de los indicadores de las Razones de Endeudamiento

Clasificación	Cálculo
Razón de Endeudamiento	Pasivo Total/Activo Total
Razón de Autonomía	Patrimonio/Activo Total

Anexo 1.3: Despliegue de los indicadores de las Razones de Actividad

Clasificación	Cálculo
Rotación del Capital Trabajo	Ventas Netas/Capital de Trabajo
Rotación de Activos	Ventas Netas /Activo Total Promedio
Rotación de Activos Fijos	Ventas Netas /Activo Fijo Tangible
Rotación de Activo Circulante	Ventas Netas/Activo Circulante
Rotación de Inventarios	$(\text{Inventarios Promedios} / \text{Costo Ventas}) * \text{cantidad días}$
Rotación de Cobro	$(\text{Cuentas por cobrar promedio} / \text{Ventas netas}) * \text{cantidad días}$
Rotación de Pago	$(\text{Cuentas por pagar promedio} / \text{Costo de ventas}) * \text{cantidad de días}$

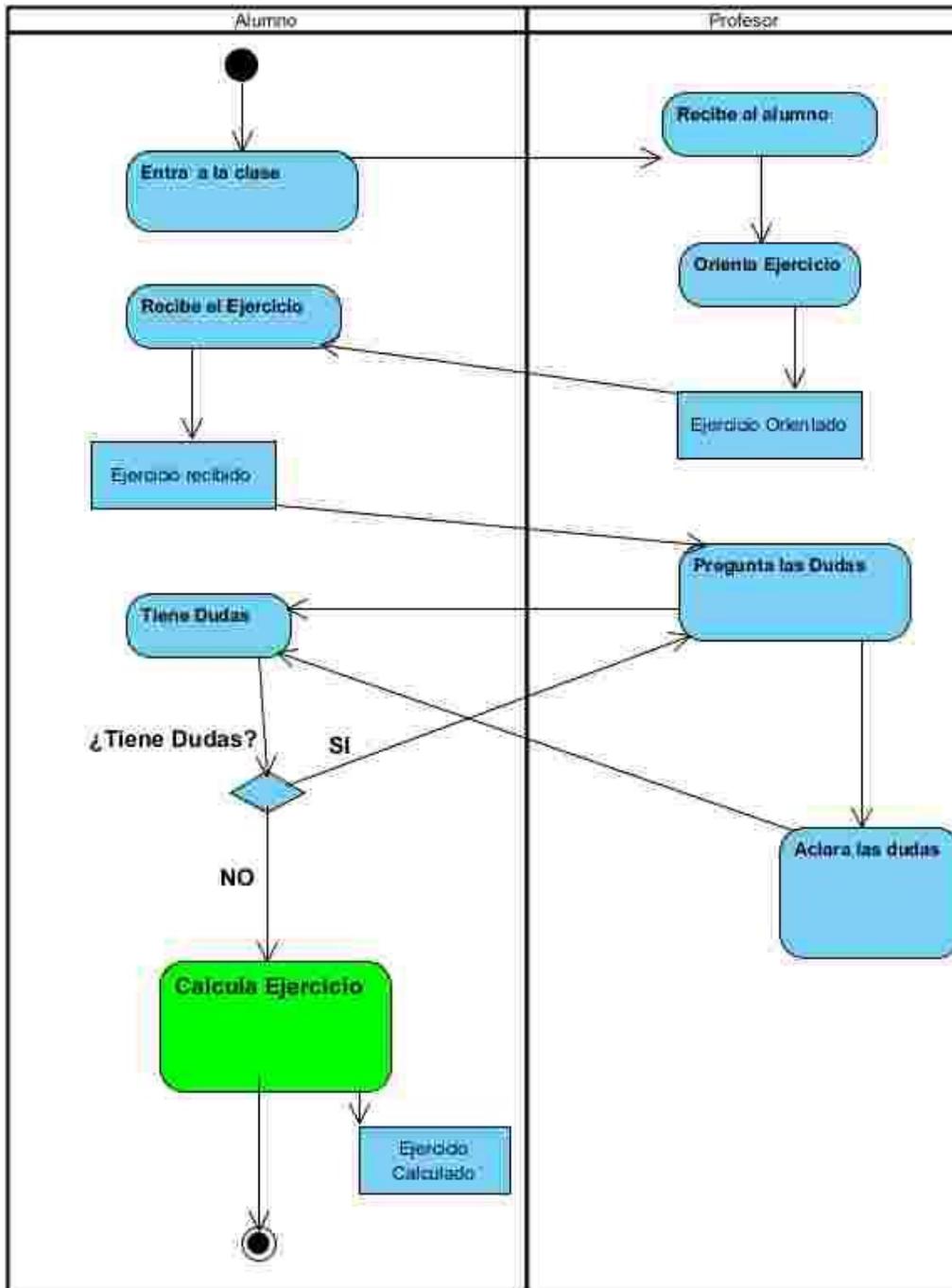
Anexo 1.4: Despliegue de los indicadores de las Razones de Rentabilidad

Clasificación	Cálculo
Rentabilidad sobre las Ventas	Utilidad Neta del Período/Ventas Netas
Rentabilidad Económica	Utilidad Neta/Activo Total
Rentabilidad Financiera	Utilidad Neta/Patrimonio
Costo por peso de Venta	Costo Ventas/Ventas Netas
Gastos de Operaciones por peso de Venta	Gastos de Operación/Ventas Netas

Anexo 1.5: Despliegue de los indicadores del Capital de Trabajo

Clasificación	Calculo
Capital de Trabajo	Activo Circulante – Pasivo Circulante
Ciclo del Efectivo	Rotación de Inventarios + Rotación de Cobro-Rotación de Pago
Rendimiento del Capital de Trabajo	Utilidad Neta del Período/Capital de Trabajo
Capital de Trabajo Necesario	(Total gastos que Originan Desembolsos/cantidad de días)*ciclo del efectivo

Anexo 2: Diagramas de actividades.



Anexo 3: Descripción de los casos del sistema.

Anexo 3.1: Descripción del caso de uso “Gestionar Activos Circulantes”

Caso de uso	Gestionar Activos Circulantes
Actores	Alumno
Propósito	Gestionar las cuentas de Activos Circulantes en el sistema.
Resumen:	El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna cuenta de Activo Circulante en el sistema.
Referencias	RF 10
Prototipo	

Anexo 3.2: Descripción del caso de uso “Gestionar Activos A Largo Plazo”

Caso de uso	Gestionar Activos A Largo Plazo
Actores	Alumno
Propósito	Gestionar las cuentas de los Activos A Largo Plazo en el sistema.
Resumen:	El caso de uso se inicia cuando el Alumno desea gestionar (Insertar, eliminar, modificar) alguna cuenta de Activo A Largo Plazo en el sistema.
Referencias	RF 11
Prototipo	

Anexo 3.3: Descripción del caso de uso “Gestionar Activos Fijos”

Caso de uso	Gestionar Activos Fijos
Actores	Alumno
Propósito	Gestionar las cuentas de los Activos Fijos en el sistema.
Resumen:	El caso de uso se inicia cuando el Alumno desea gestionar (Insertar, eliminar, modificar) alguna cuenta de Activo Fijo en el sistema.
Referencias	RF 12
Prototipo	

Anexo 3.4: Descripción del caso de uso “Gestionar Activos Fijos”

Caso de uso	Gestionar Activos Diferidos
Actores	Alumno
Propósito	Gestionar las cuentas de Activos Diferidos en el sistema.
Resumen:	El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna cuenta de Activo Diferido en el sistema.
Referencias	RF 13
Prototipo	

Anexo 3.5: Descripción del caso de uso “Gestionar Otros Activos”

Caso de uso	Gestionar Otros Activos
Actores	Alumno
Propósito	Gestionar las cuentas de Otros Activos en el sistema.
Resumen:	El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna cuenta de Otros Activos en el sistema.
Referencias	RF 14
Prototipo	

Anexo 3.6: Descripción del caso de uso “Gestionar Cuentas Reguladoras de Activos”

Caso de uso	Gestionar Cuentas Reguladoras de Activos
Actores	Alumno
Propósito	Gestionar las Cuentas Reguladoras de Activos en el sistema.
Resumen:	El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna cuenta Reguladora de Activo en el sistema.
Referencias	RF 15
Prototipo	

Anexo 3.7: Descripción del caso de uso “Gestionar Pasivos Circulantes”

Caso de uso	Gestionar Pasivos Circulantes
Actores	Alumno
Propósito	Gestionar las cuentas de los Pasivos Circulantes en el sistema.
Resumen: El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna cuenta del Pasivo Circulante en el sistema.	
Referencias	RF 16
Prototipo	

Anexo 3.8: Descripción del caso de uso “Gestionar Pasivos A Largo Plazo”

Caso de uso	Gestionar Pasivos A Largo Plazo
Actores	Alumno
Propósito	Gestionar las cuentas de los Pasivos A Largo Plazo en el sistema.
Resumen: El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna cuenta del Pasivo a Largo Plazo en el sistema.	
Referencias	RF 17
Prototipo	

Anexo 3.9: Descripción del caso de uso “Gestionar Pasivos Diferidos”

Caso de uso	Gestionar Pasivos Diferidos
Actores	Alumno
Propósito	Gestionar las cuentas de los Pasivos Diferidos en el sistema.
Resumen: El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna cuenta del Pasivo Diferido en el sistema.	
Referencias	RF 18
Prototipo	

Anexo 3.10: Descripción del caso de uso “Gestionar Otros Pasivos”

Caso de uso	Gestionar Otros Pasivos
Actores	Alumno
Propósito	Gestionar las cuentas de Otros Pasivos en el sistema.
Resumen: El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna cuenta de Otro Pasivo en el sistema.	
Referencias	RF 19
Prototipo	

Anexo 3.11: Descripción del caso de uso “Gestionar Patrimonio Neto o Capital Contable”.

Caso de uso	Gestionar Patrimonio Neto o Capital Contable.
Actores	Alumno
Propósito	Gestionar las cuentas del Patrimonio Neto o Capital Contable (sector público) en el sistema.
Resumen: El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna cuenta del Patrimonio Neto o Capital Contable (sector público) en el sistema.	
Referencias	RF 20
Prototipo	

Anexo 3.12: Descripción del caso de uso “Gestionar Cuentas de Compras”.

Caso de uso	Gestionar Cuentas de Compras
Actores	Alumno
Propósito	Gestionar las cuentas de Compras en el sistema.
Resumen: El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna cuenta de Compras en el sistema.	
Referencias	RF 21
Prototipo	

Anexo 3.13: Descripción del caso de uso “Gestionar Gastos de Producción”

Caso de uso	Gestionar Gastos de Producción
Actores	Alumno
Propósito	Gestionar las cuentas de los Gastos de Producción en el sistema.
Resumen:	El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna cuenta de los Gastos de Producción en el sistema.
Referencias	RF 22
Prototipo	

Anexo 3.14: Descripción del caso de uso “Gestionar Cuentas Nominales Deudoras (Excepto Empresas de Seguro)”

Caso de uso	Gestionar Cuentas Nominales Deudoras (Excepto Empresas de Seguro).
Actores	Alumno
Propósito	Gestionar las cuentas Nominales Deudoras (Excepto Empresas de Seguro) en el sistema.
Resumen:	El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna de las cuentas Nominales Deudoras (Excepto Empresas de Seguro) en el sistema.
Referencias	RF 23
Prototipo	

Anexo 3.15: Descripción del caso de uso “Gestionar Cuentas Nominales Acreedoras (Excepto Empresas de Seguro)”.

Caso de uso	Gestionar Cuentas Nominales Acreedoras (Excepto Empresas de Seguro).
Actores	Alumno
Propósito	Gestionar las cuentas Acreedoras (Excepto Empresas de Seguro) en el sistema.
Resumen: El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna de las cuentas Nominales Acreedoras (Excepto Empresas de Seguro) en el sistema.	
Referencias	RF 24
Prototipo	

Anexo 3.16: Descripción del caso de uso “Gestionar Cuentas Nominales Deudoras (Empresas de Seguro)”.

Caso de uso	Gestionar Cuentas Nominales Deudoras (Empresas de Seguro).
Actores	Alumno
Propósito	Gestionar las cuentas Nominales Deudoras (Empresas de Seguro) en el sistema.
Resumen: El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna de las cuentas Nominales Deudoras (Empresas de Seguro) en el sistema.	
Referencias	RF 25
Prototipo	

Anexo 3.17: Descripción del caso de uso “Gestionar Cuentas Nominales Acreedoras (Empresas de Seguro)”.

Caso de uso	Gestionar Cuentas Nominales Acreedoras (Empresas de Seguro)
Actores	Alumno
Propósito	Gestionar las cuentas Nominales Acreedoras (Empresas de Seguro) en el sistema.
Resumen: El caso de uso se inicia cuando el Alumno desea gestionar (insertar, eliminar, modificar) alguna de las cuentas Nominales Acreedoras (Empresas de Seguro) en el sistema.	
Referencias	RF 26
Prototipo	

Anexo 3.18: Descripción del caso de uso “Mostrar Reporte de las Razones de Liquidez”.

Caso de uso	Mostrar Reporte de las Razones de Liquidez.
Actores	Alumno
Propósito	Conocer la Razón de Liquidez con que cuenta la Entidad para un período dado.
Resumen: Comienza cuando el Usuario accede al menú de Razones de Liquidez y se le pide seleccionar los años para los cuales desea mostrar las Razones de Liquidez en caso de existir la fecha solicitada se genera el reporte y se muestra, en caso contrario se le informa al usuario de que la fecha no existe en la base de datos. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF 3
Prototipo	

Anexo 3.19: Descripción del caso de uso “Mostrar Reporte de las Razones de Solvencia”.

Caso de uso	Mostrar Reporte de las Razones de Solvencia.
Actores	Alumno
Propósito	Conocer las Razones de Solvencia con que cuenta la Entidad para un período dado.
Resumen: Comienza cuando el Usuario accede al menú de Razones de Solvencia y se le pide seleccionar los años para los cuales desea mostrar las Razones de Solvencia en caso de existir la fecha solicitada se genera el reporte y se muestra, en caso contrario se le informa al usuario de que la fecha no existe en la base de datos. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF 4
Prototipo	

Anexo 3.20: Descripción del caso de uso “Mostrar Reporte de la Razones de Solidez”.

Caso de uso	Mostrar Reporte de las Razones de Solidez.
Actores	Alumno
Propósito	Conocer las Razones de Solidez con que cuenta la Entidad para un período dado.
Resumen: Comienza cuando el Usuario accede al menú de Razones de Solidez y se le pide seleccionar los años para los cuales desea mostrar las Razones de Solidez en caso de existir la fecha solicitada se genera el reporte y se muestra, en caso contrario se le informa al usuario de que la fecha no existe en la base de datos. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF 5
Prototipo	

Anexo 3.21: Descripción del caso de uso “Mostar Reporte de las Razones de Endeudamiento”.

Caso de uso	Mostar Reporte de las Razones de Endeudamiento.
Actores	Alumno
Propósito	Conocer las Razones de Endeudamiento con que cuenta la Entidad para un período dado.
Resumen: Comienza cuando el Usuario accede al menú de Razones de Endeudamiento y se le pide seleccionar los años para los cuales desea mostrar las Razones de Endeudamiento en caso de existir la fecha solicitada se genera el reporte y se muestra, en caso contrario se le informa al usuario de que la fecha no existe en la base de datos. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF 6
Prototipo	

Anexo 3.22: Descripción del caso de uso “Mostar Reporte de la Razones de Rotación”.

Caso de uso	Mostar Reporte de las Razones de Rotación.
Actores	Alumno
Propósito	Conocer las Razones de Rotación con que cuenta la Entidad para un período dado.
Resumen: Comienza cuando el Usuario accede al menú de Razones de Rotación y se le pide seleccionar los años para los cuales desea mostrar las Razones de Rotación en caso de existir la fecha solicitada se genera el reporte y se muestra, en caso contrario se le informa al usuario de que la fecha no existe en la base de datos. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.	
Referencias	RF 7
Prototipo	

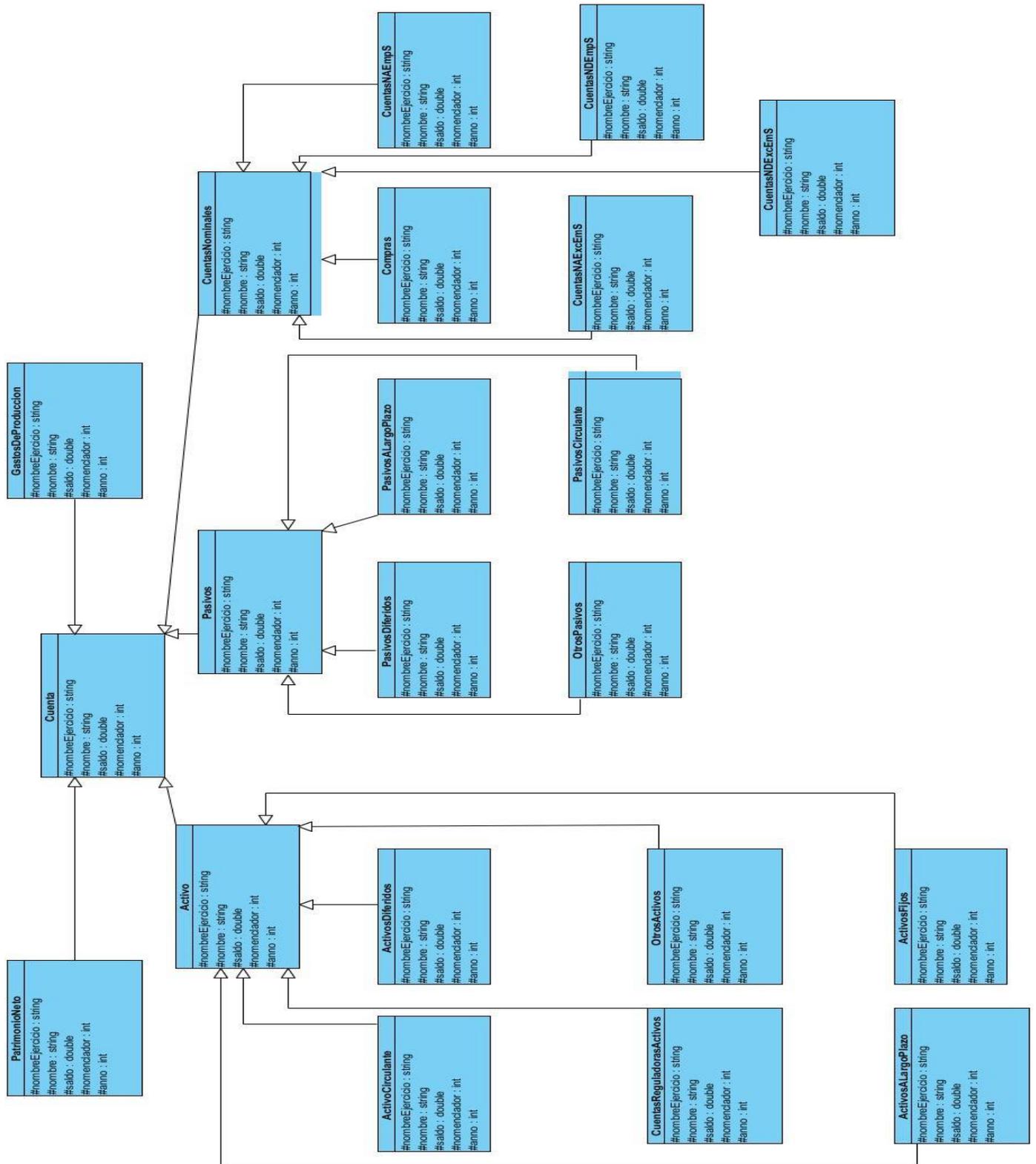
Anexo 3.23: Descripción del caso de uso “Mostrar Reportes de las Razones de Rentabilidad”.

Caso de uso	Mostrar Reportes de las Razones de Rentabilidad.
Actores	Alumno
Propósito	Conocer las Razones de Rentabilidad con que cuenta la Entidad para un período dado.
<p>Resumen: Comienza cuando el Usuario accede al menú de Razones de Rentabilidad y se le pide seleccionar los años para los cuales desea mostrar las Razones de Rentabilidad en caso de existir la fecha solicitada se genera el reporte y se muestra, en caso contrario se le informa al usuario de que la fecha no existe en la base de datos.</p> <p>El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.</p>	
Referencias	RF 8
Prototipo	

Anexo 3.24: Descripción del caso de uso “Mostrar Reporte sobre Ventas”.

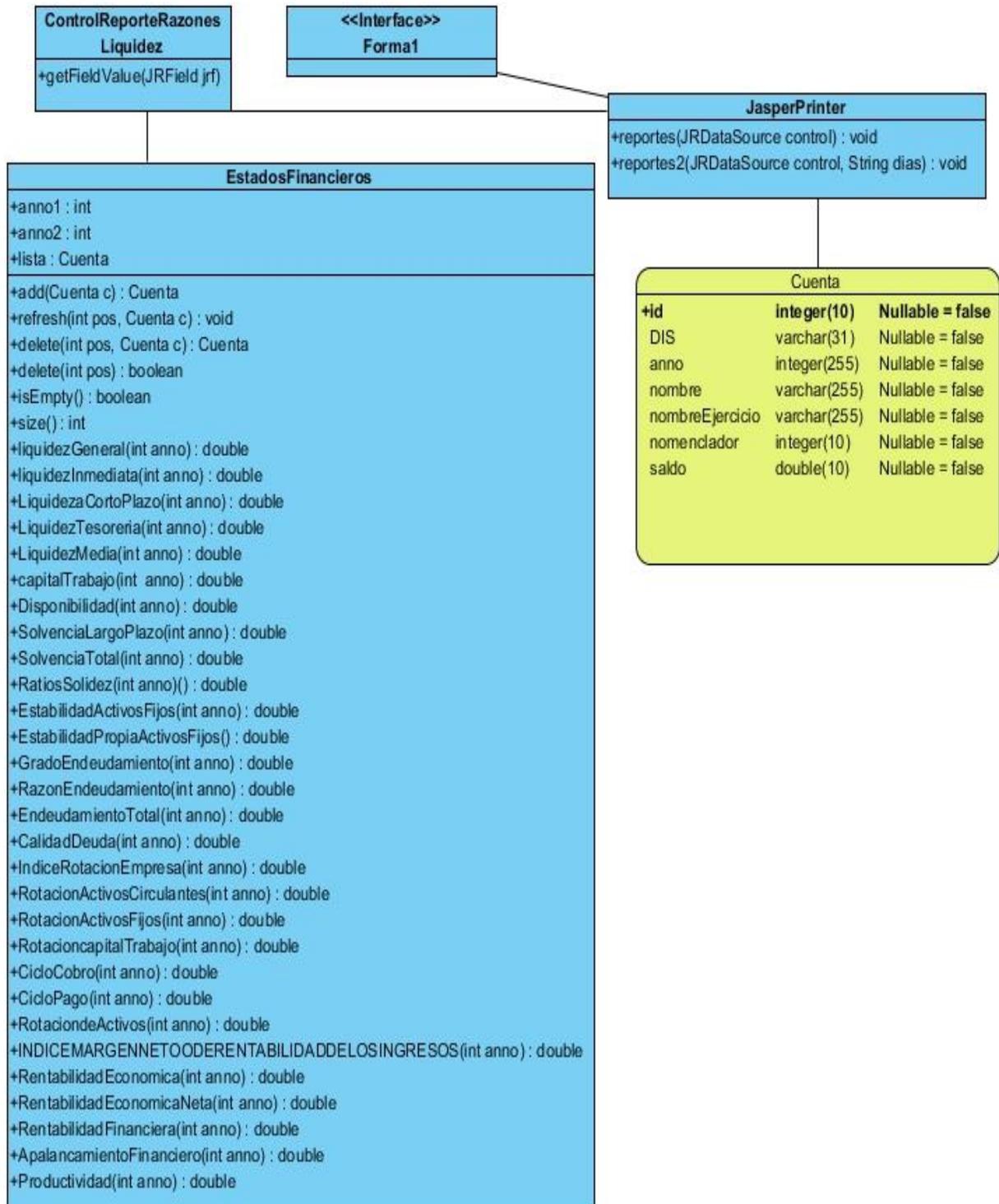
Caso de uso	Mostrar Reporte sobre Ventas
Actores	Alumno
Propósito	Conocer los Reportes sobre Ventas con que cuenta la Entidad para un período dado
<p>Resumen: Comienza cuando el Usuario accede al menú Reportes sobre Ventas y se le pide seleccionar los años para los cuales desea mostrar los Reportes sobre Ventas en caso de existir la fecha solicitada se genera el reporte y se muestra, en caso contrario se le informa al usuario de que la fecha no existe en la base de datos. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descritas.</p>	
Referencias	RF 9
Prototipo	

Anexo 4: Diagrama de clases persistentes.

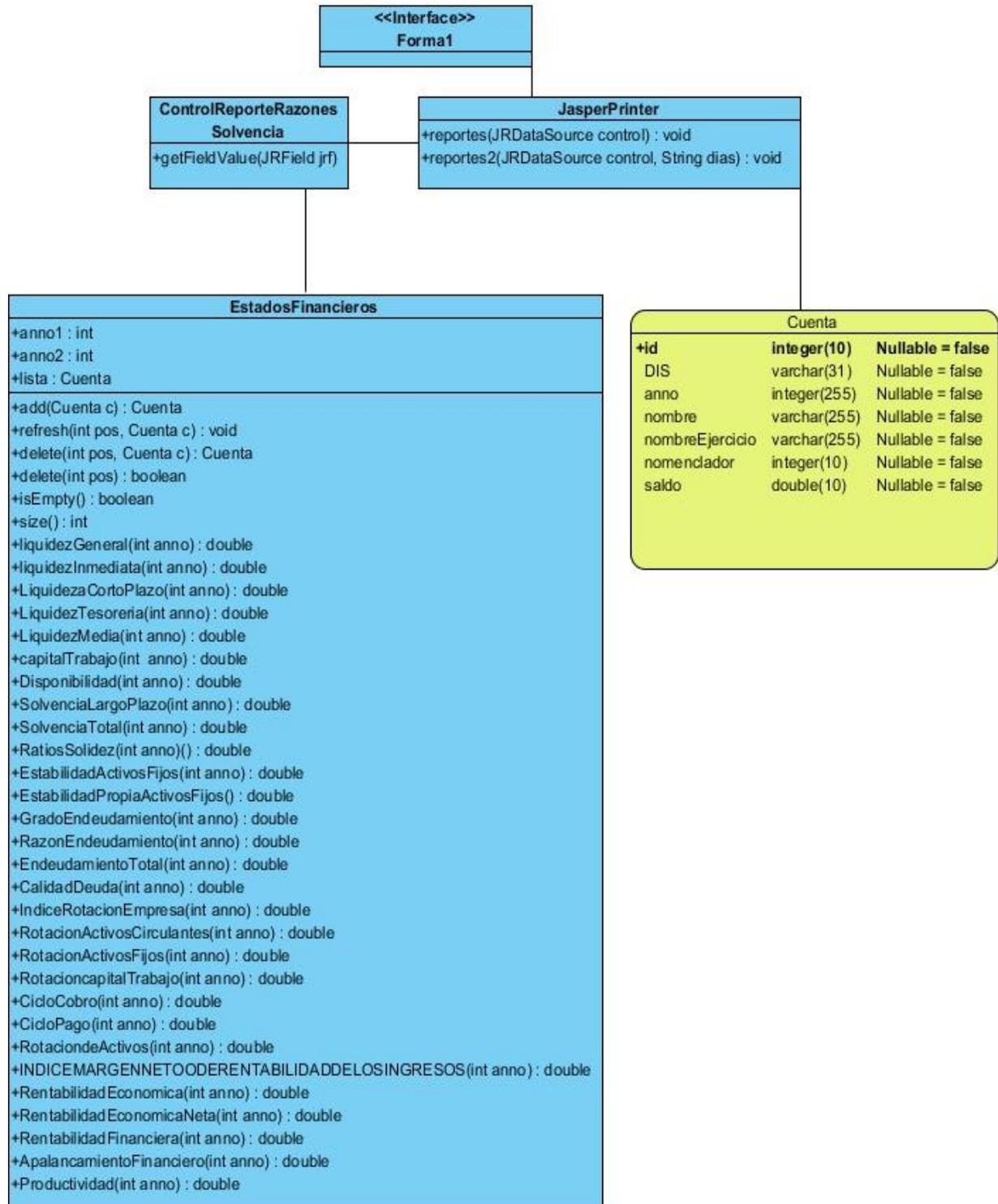


Anexo 5: Diseños de clases del sistema

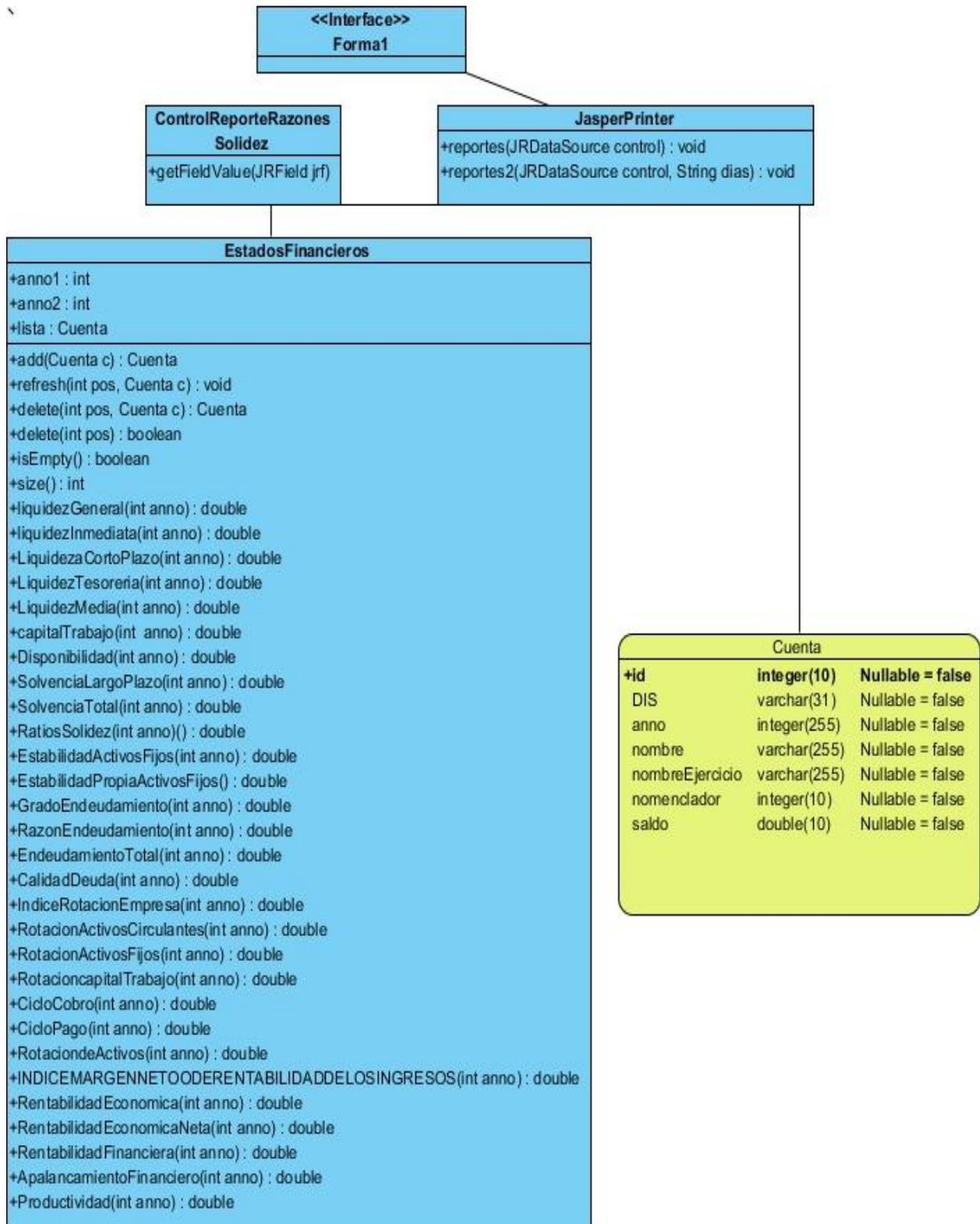
Anexo 5.1: Caso de uso Mostrar Reporte de las Razones de Liquidez.



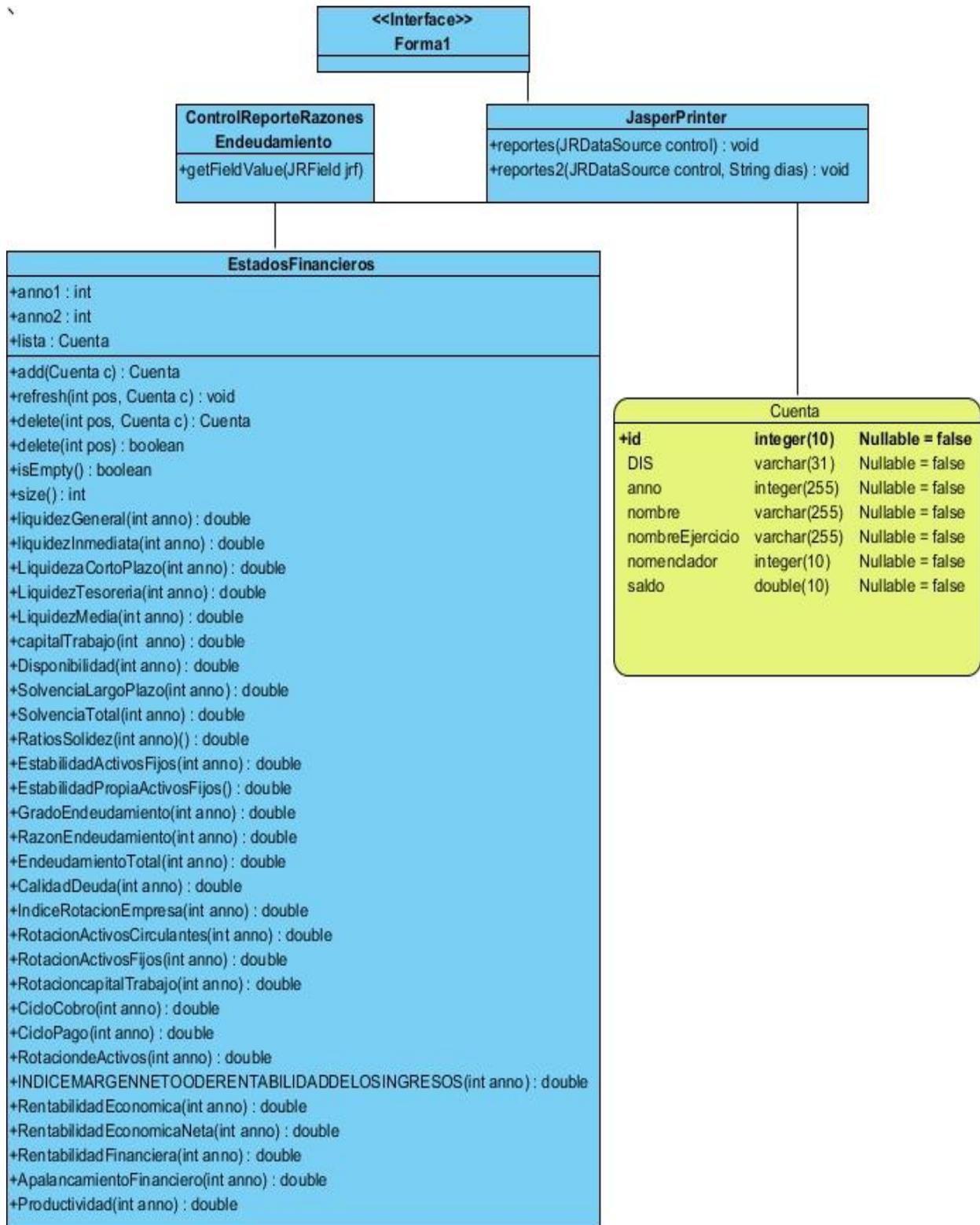
Anexo 5.2: Caso de uso Mostrar Reporte de las Razones de Solvencia.



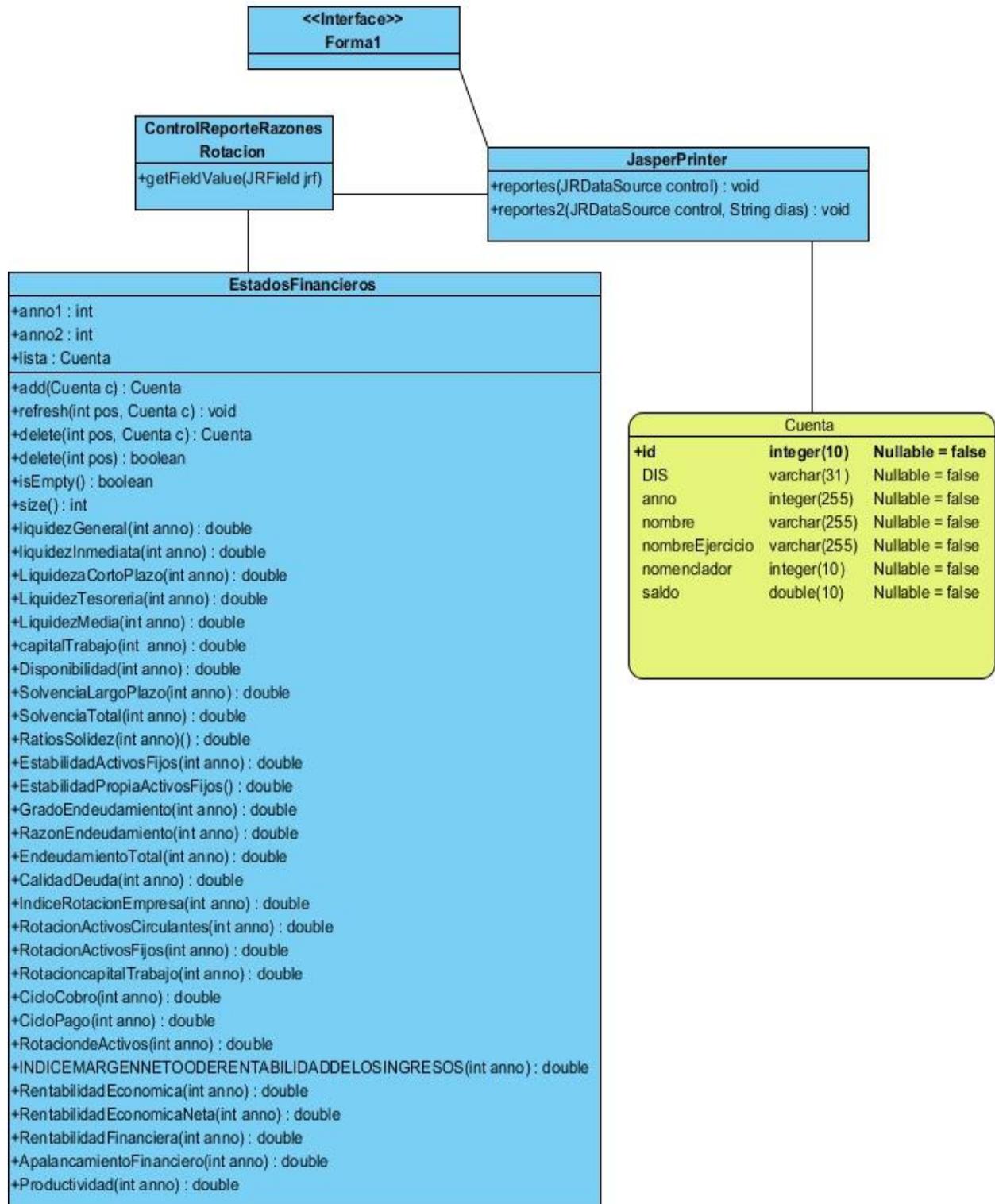
Anexo 5.3: Caso de uso Mostrar Reporte de las Razones de Solidez.



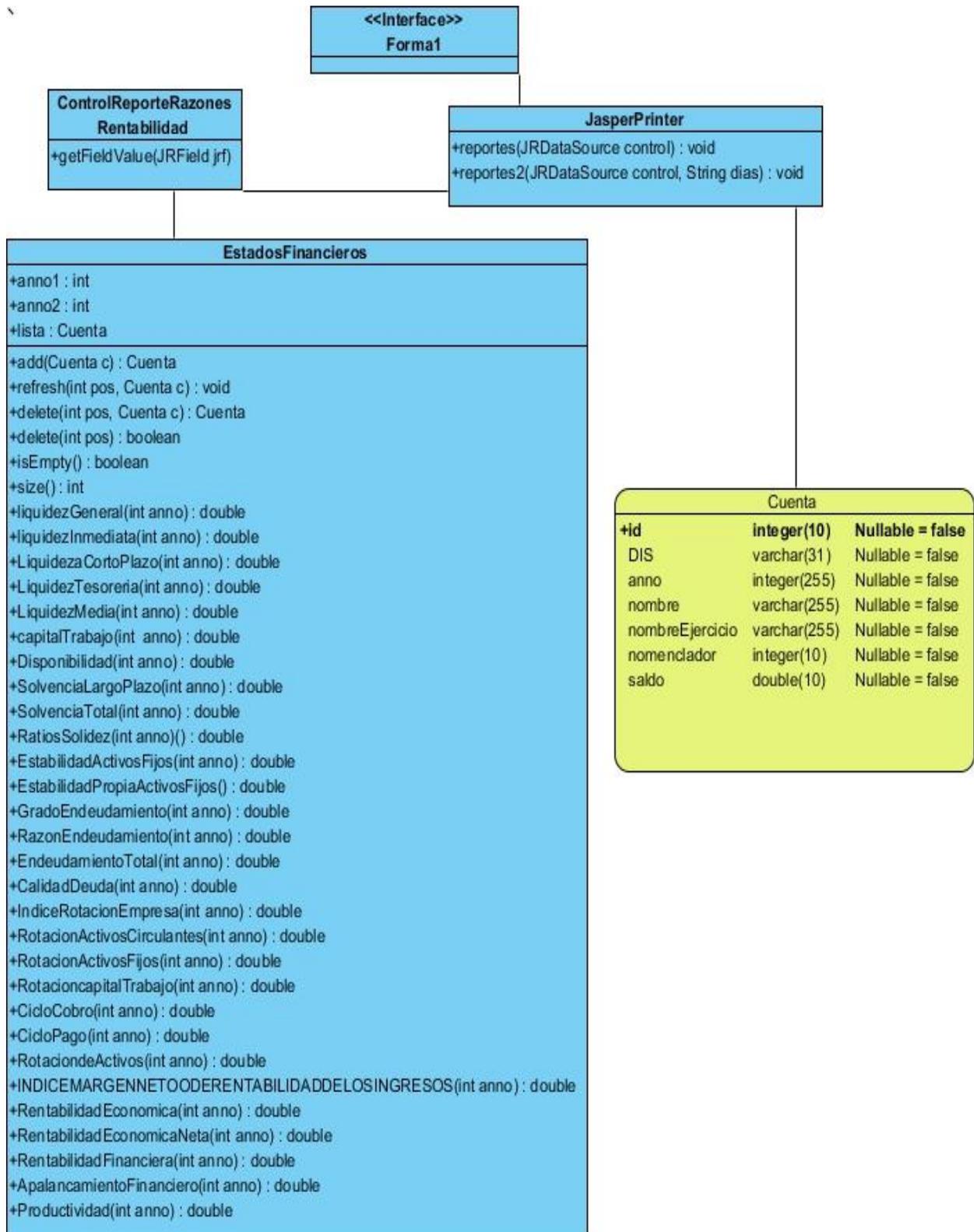
Anexo 5.4: Caso de uso Mostar Reporte de las Razones de Endeudamiento.



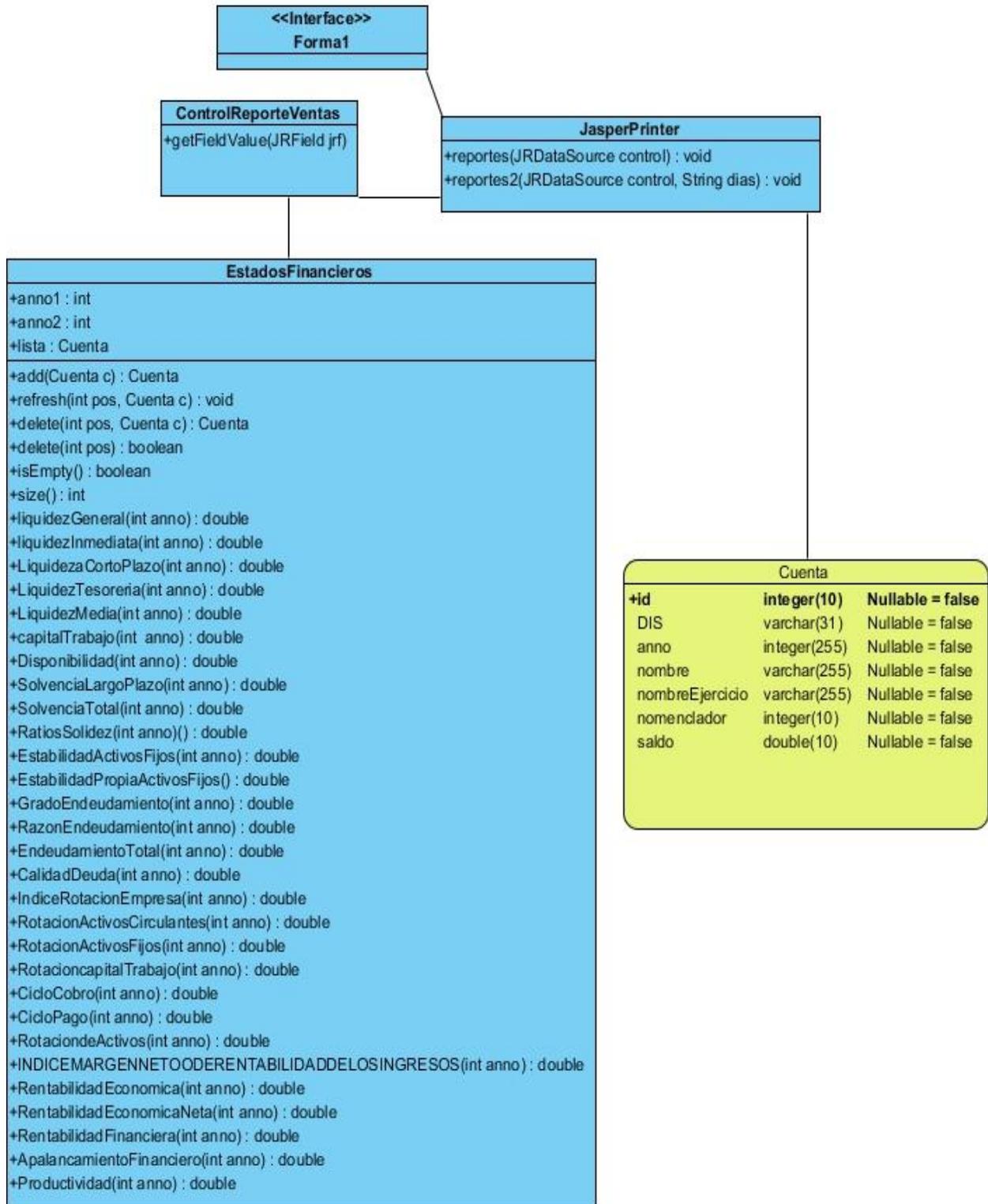
Anexo 5.5: Caso de uso Mostar Reporte de las Razones de Rotación.



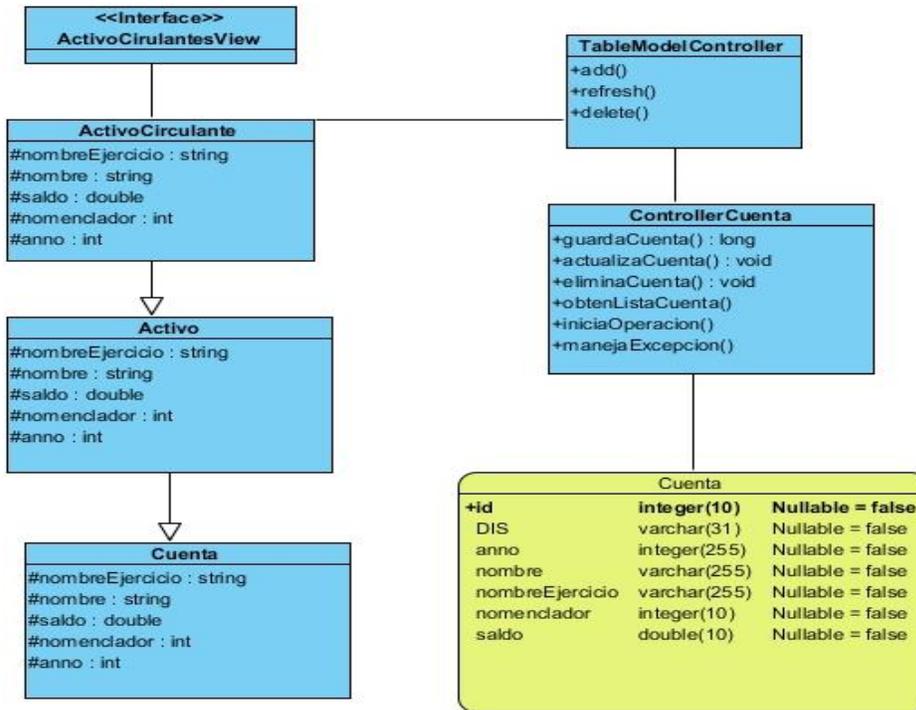
Anexo 5.6: Caso de uso Mostrar Reportes de las Razones de Rentabilidad.



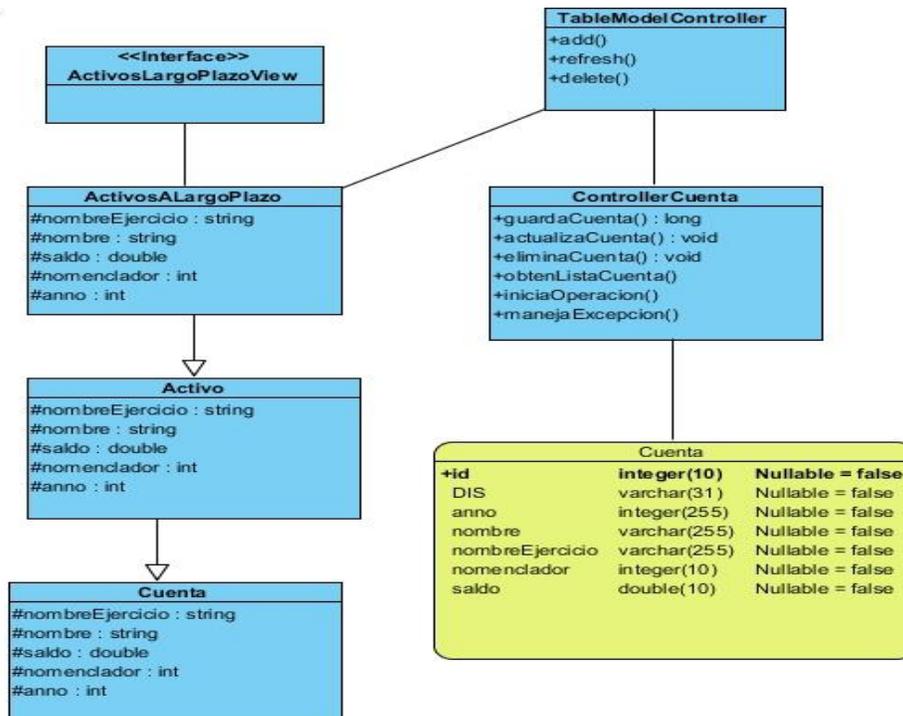
Anexo 5.7: Caso de uso Mostrar Reporte sobre Ventas.



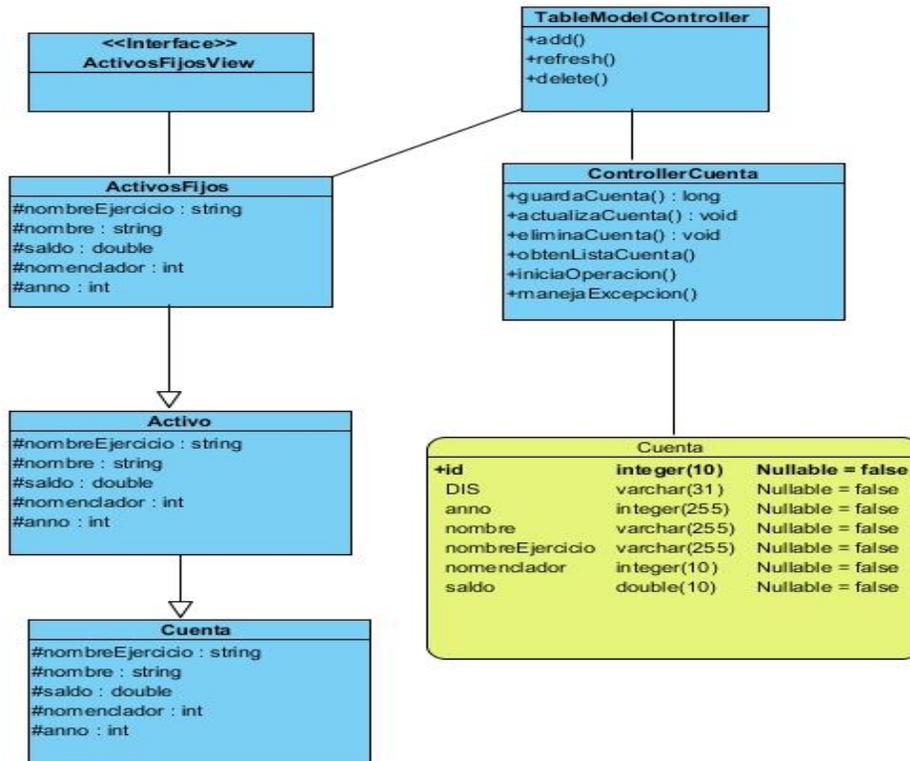
Anexo 5.8: Caso de uso Gestionar Activos Circulantes.



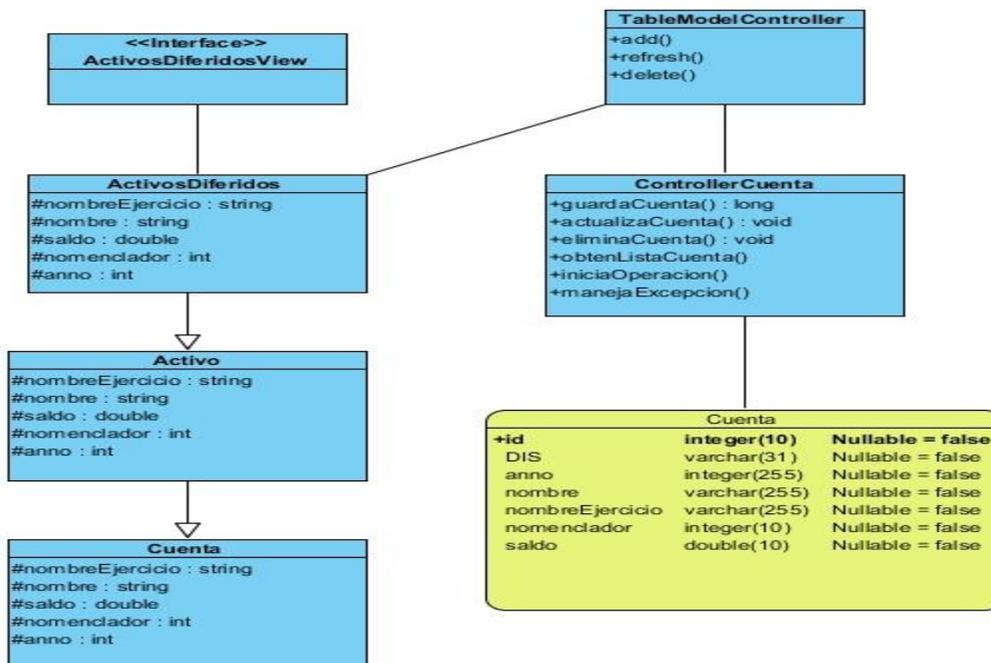
Anexo 5.9: Caso de uso Gestionar Activos A Largo Plazo.



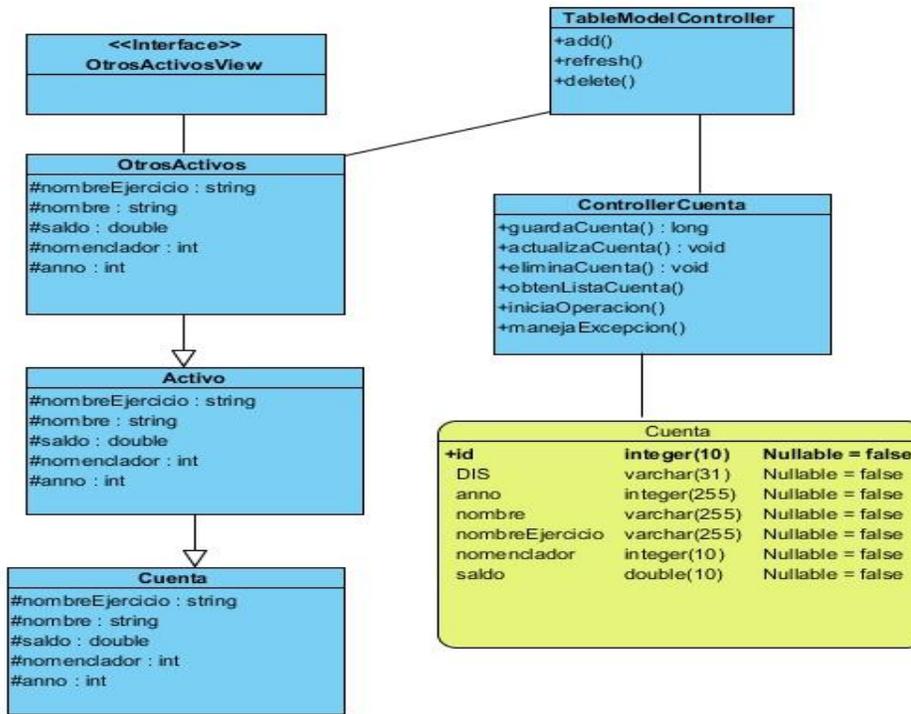
Anexo 5.10: Caso de uso Gestionar Activos Fijos.



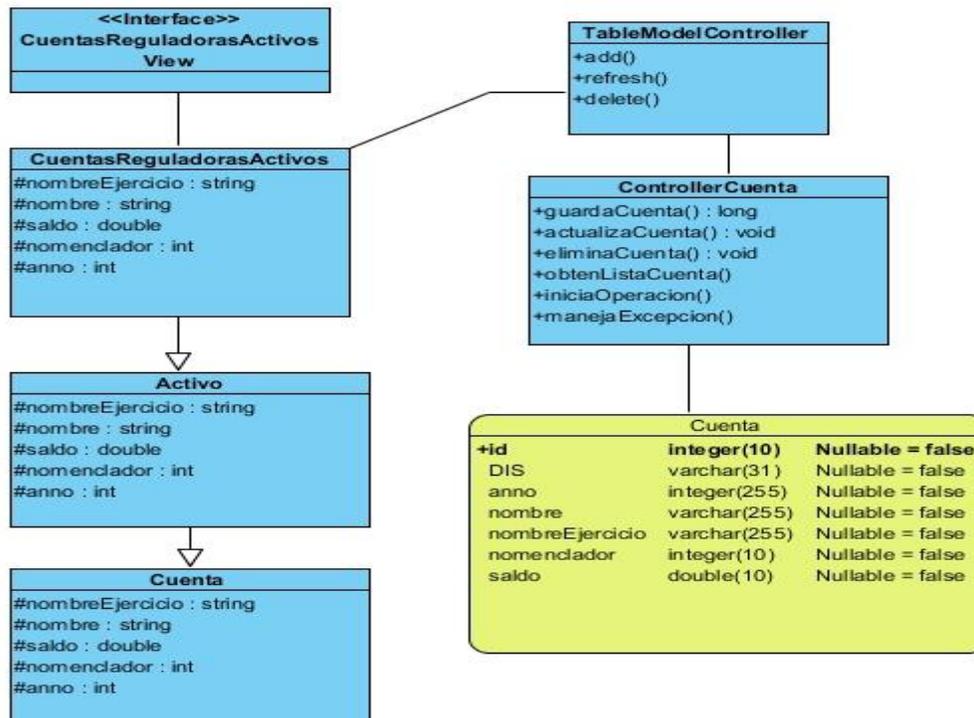
Anexo 5.11: Caso de uso Gestionar Activos Diferidos.



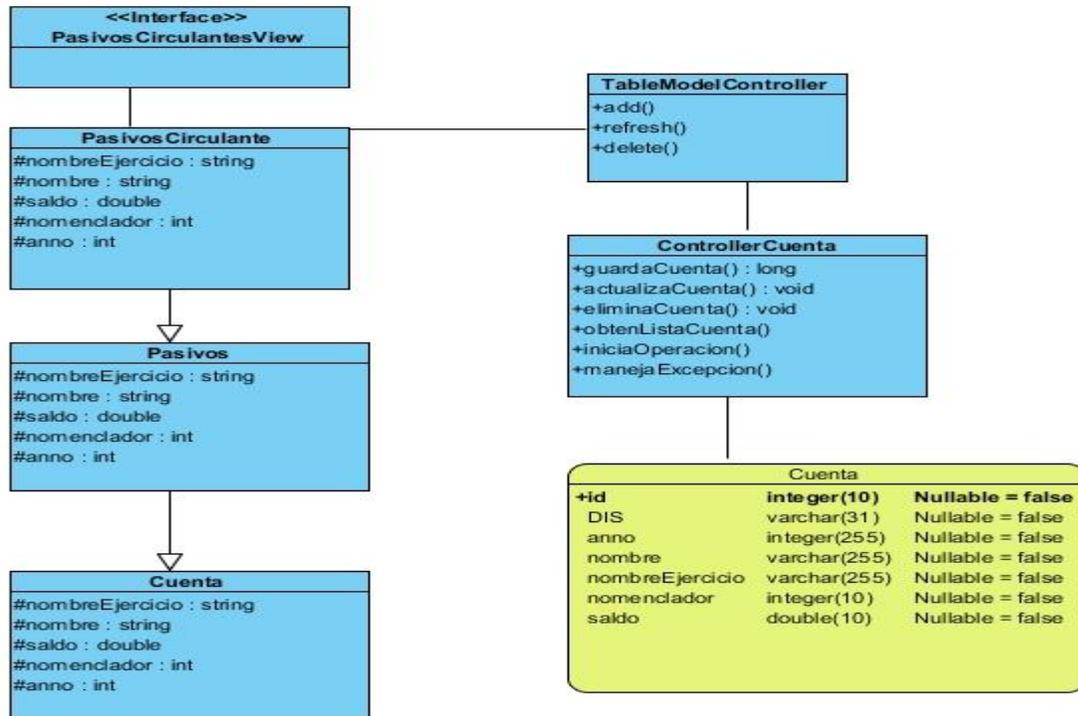
Anexo 5.12: Caso de uso Gestionar Otros Activos.



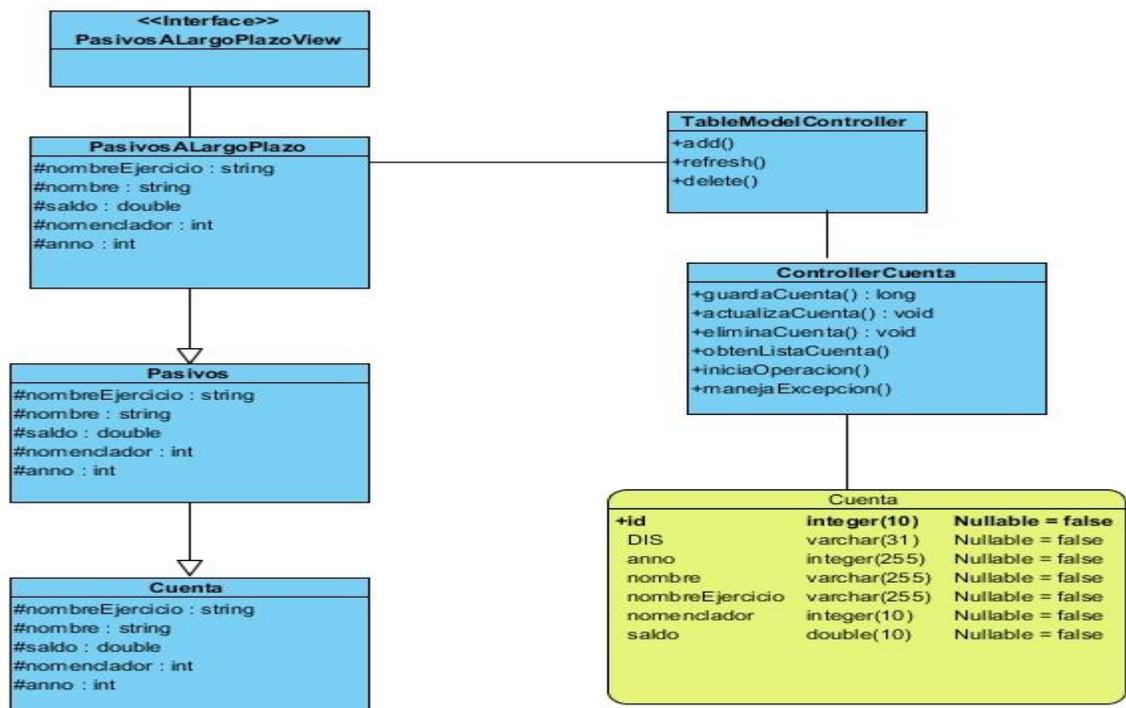
Anexo 5.13: Caso de uso Gestionar Cuentas Reguladoras de Activos.



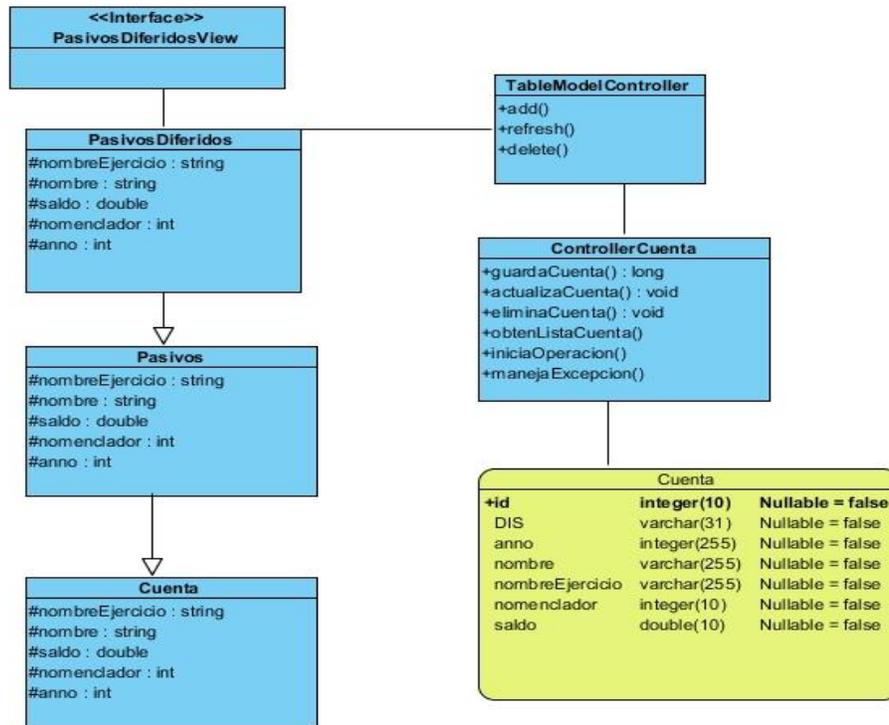
Anexo 5.14: Caso de uso Gestionar Pasivos Circulantes.



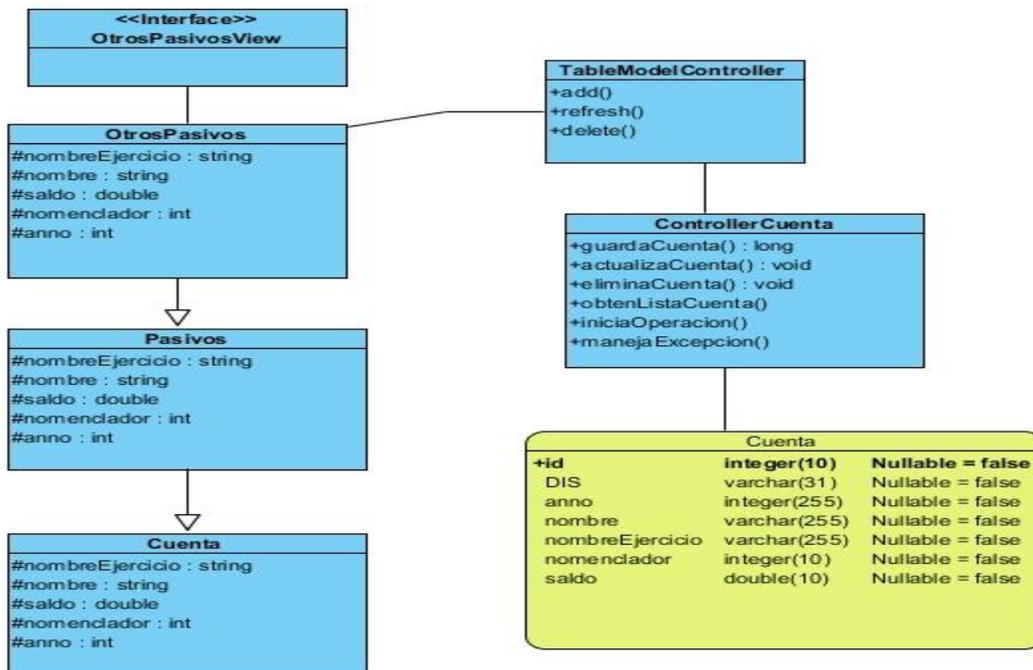
Anexo 5.15: Caso de uso Gestionar Pasivos A Largo Plazo.



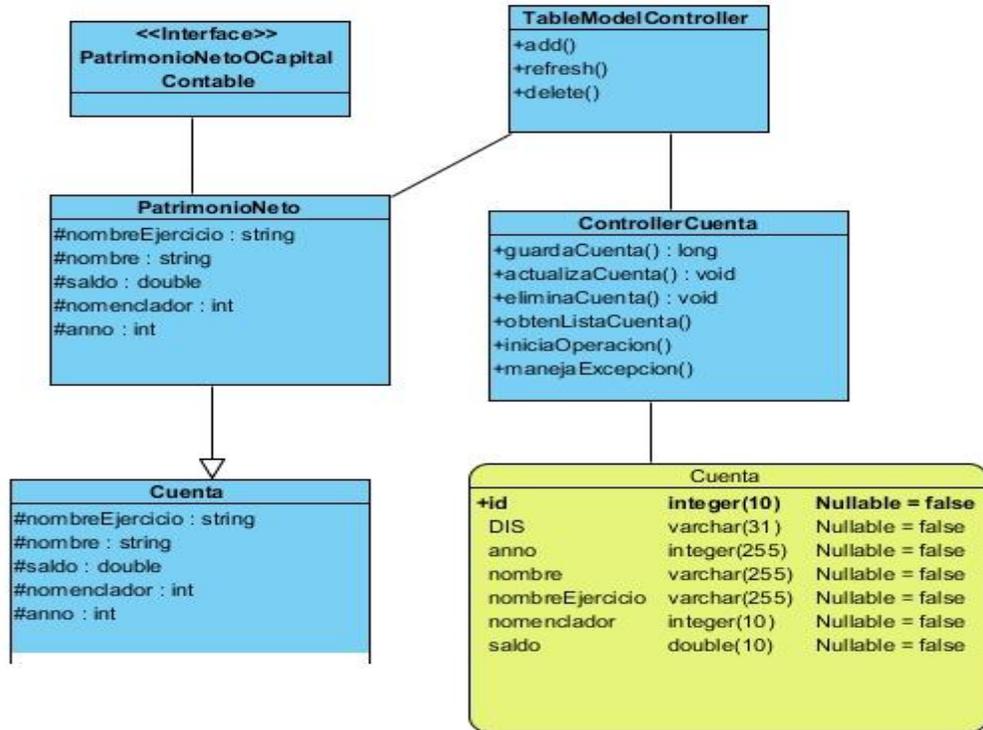
Anexo 5.16: Caso de uso Gestionar Pasivos Diferidos.



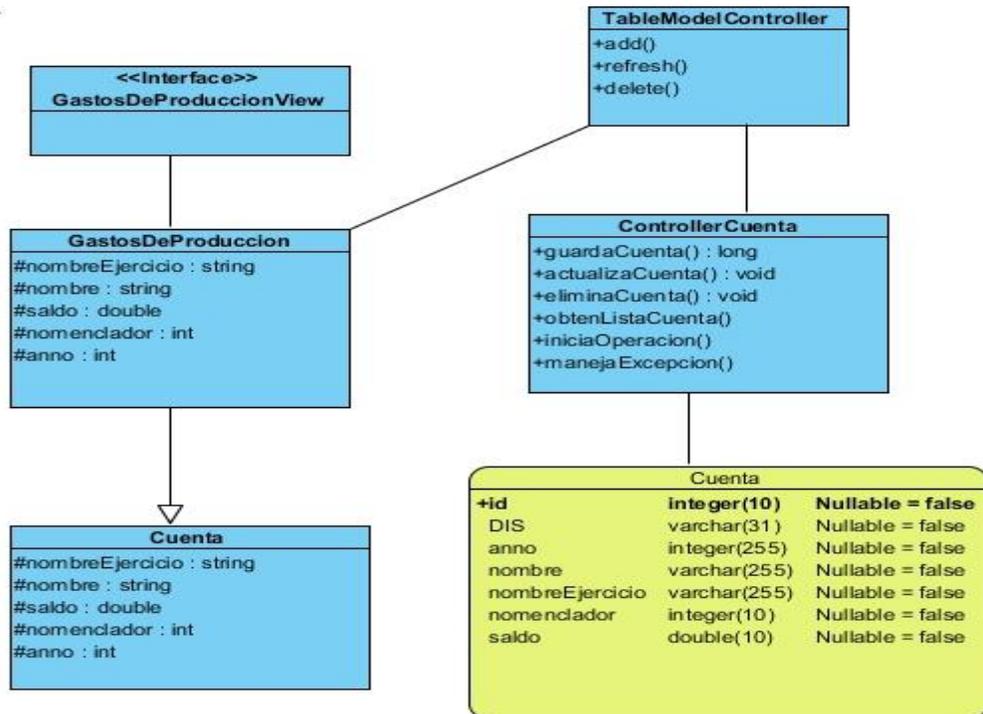
Anexo 5.17: Caso de uso Gestionar Otros Pasivos.



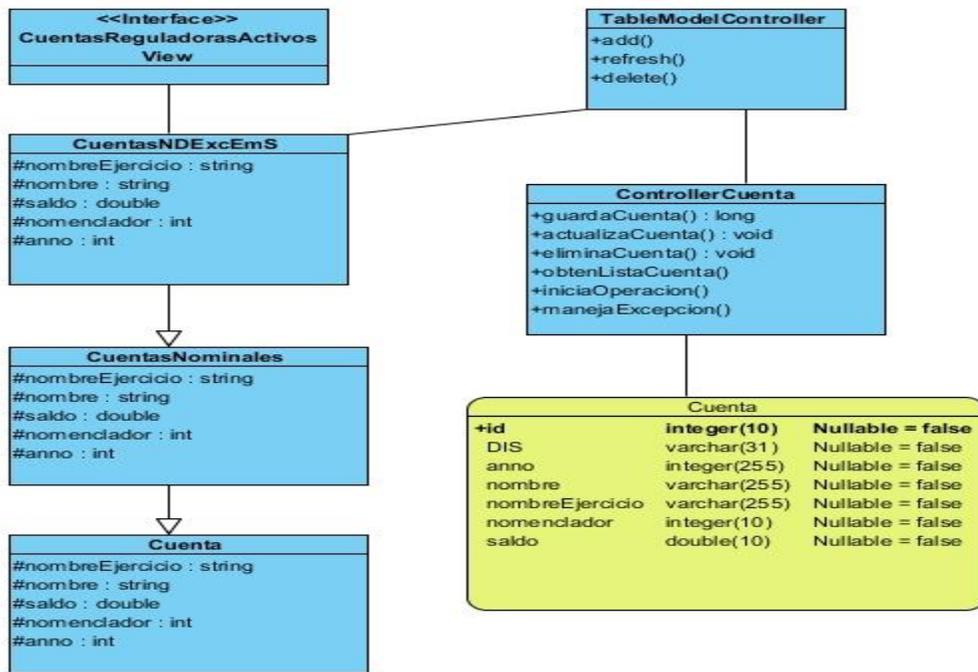
Anexo 5.18: Caso de uso Gestionar Patrimonio Neto o Capital Contable.



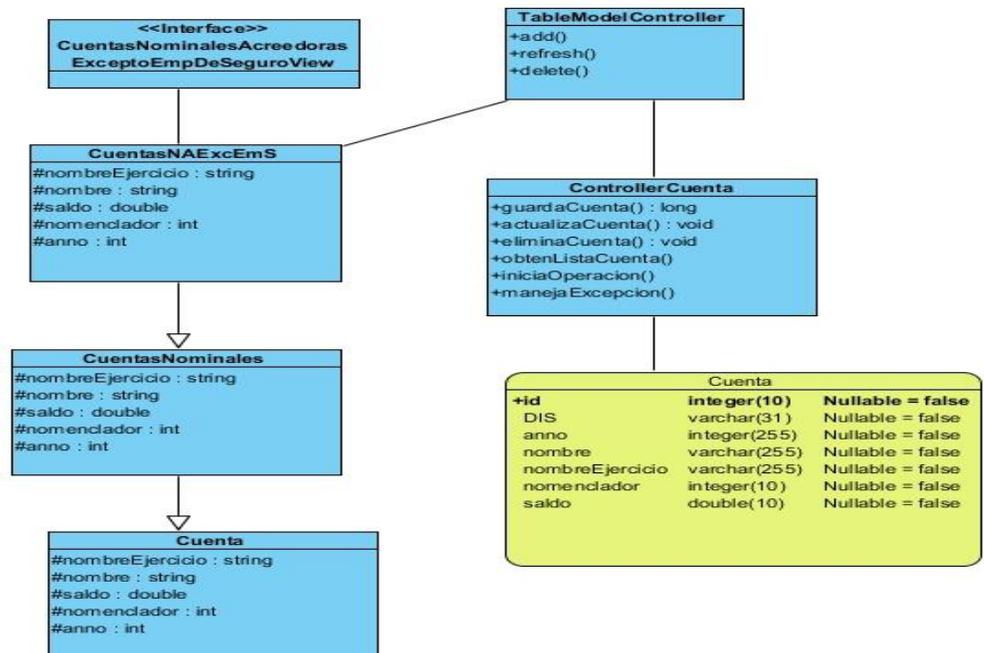
Anexo 5.19: Caso de uso Gestionar Gastos de Producción.



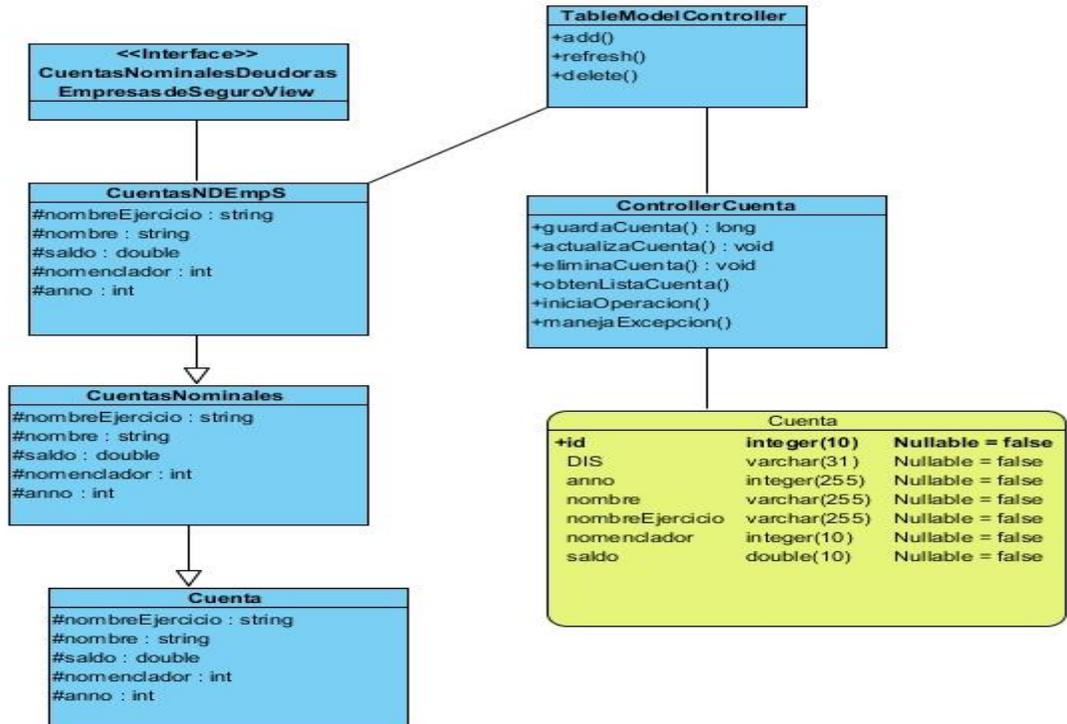
Anexo 5.20: Caso de uso Gestionar Cuentas Nominales Deudoras (Excepto Empresas de Seguro).



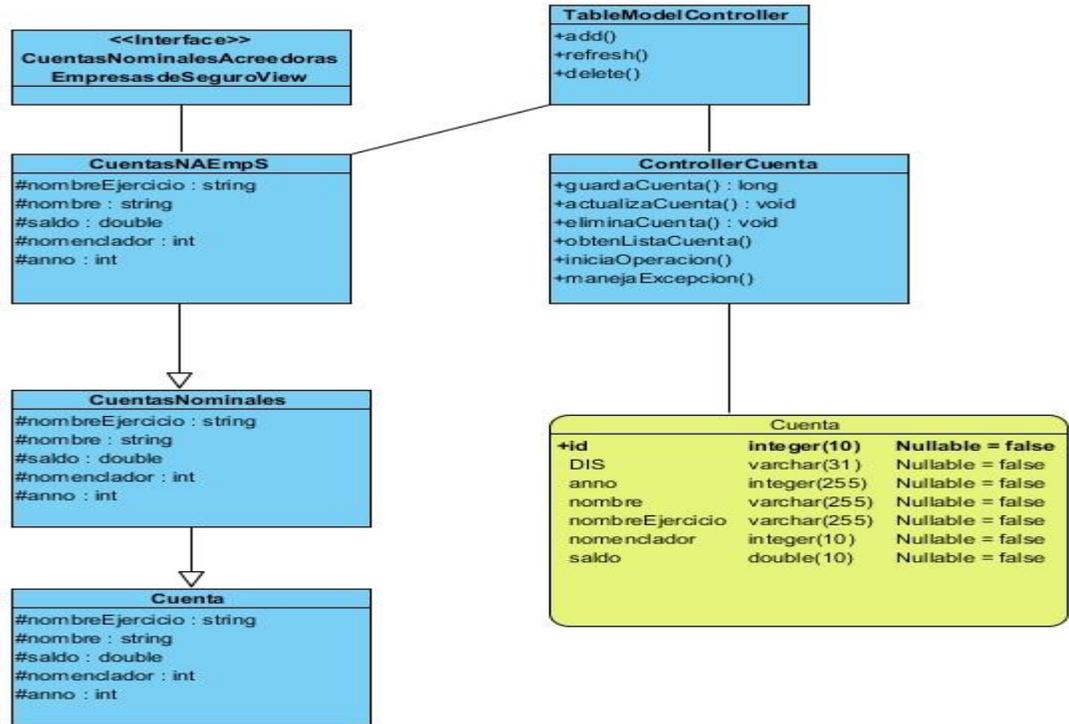
Anexo 5.21: Caso de uso Gestionar Cuentas Nominales Acreedoras (Excepto Empresas de Seguro).



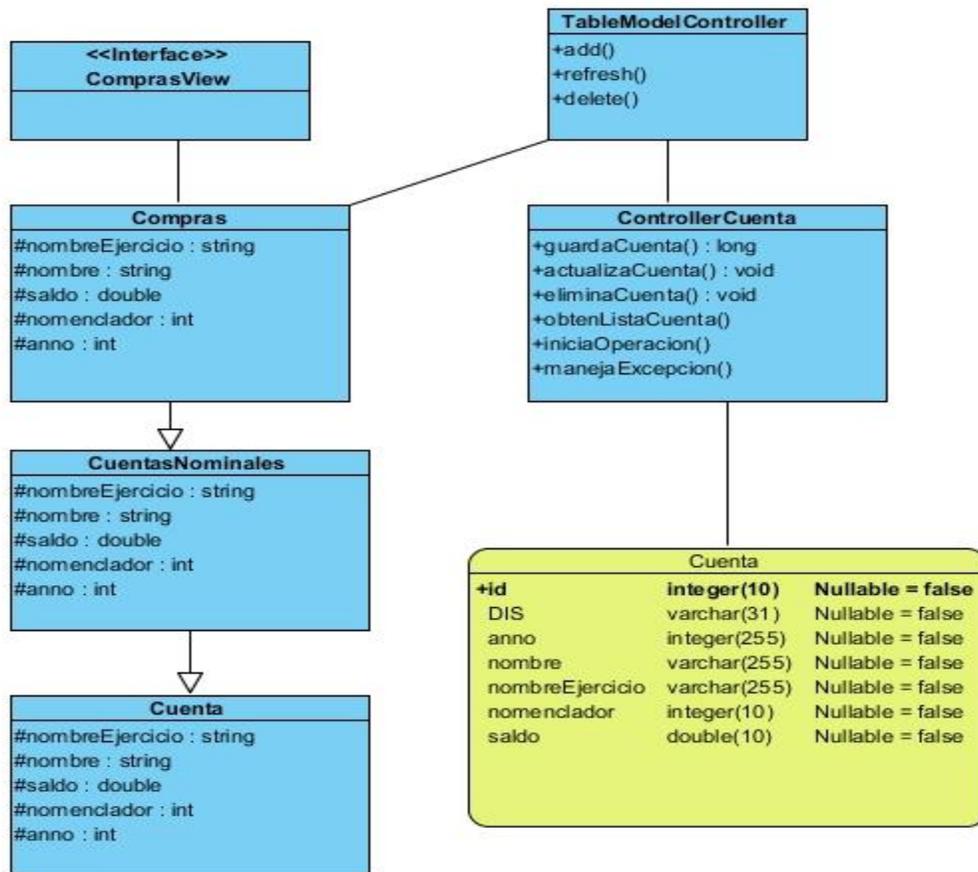
Anexo 5.22: Caso de uso Gestionar Cuentas Nominales Deudoras (Empresas de Seguro).



Anexo 5.23: Caso de uso Gestionar Cuentas Nominales Acreedoras (Empresas de Seguro).

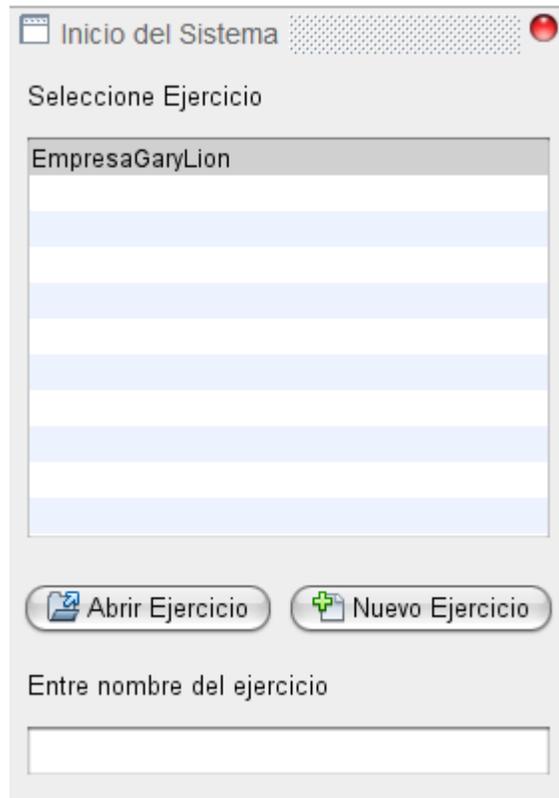


Anexo 5.24: Caso de uso Gestionar Cuentas de Compras.

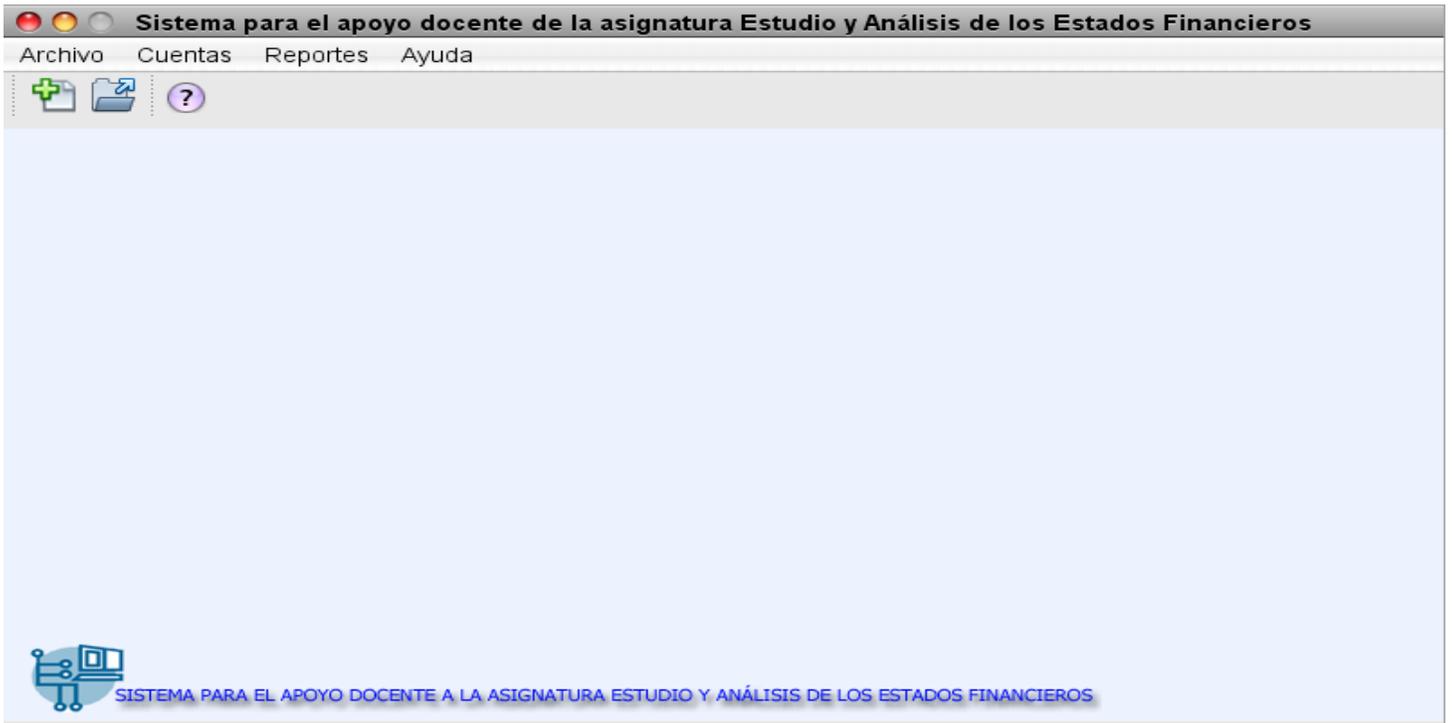


Anexo 6 Ventanas del Software.

Anexo 6.1: Ventana de Inicio del Software.



Anexo 6.2: Ventana Principal del Software



Anexo 6.3: Caso de Uso Gestionar Activos Circulantes

The screenshot shows the "Gestionar Activos Circulantes" window. It contains a table with the following data:

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SUPERFICIE P...	7854.0	126	2015
INMUEBLES PARA COMERCIA...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUSTIFICAR	7854.0	245	2014

Below the table, there is a dropdown menu with "ANTICIPOS A JUSTIFICAR" selected. Underneath, there are three input fields labeled "Entre el Saldo de la Cuenta", "Entre el Nomenclador de la Cuenta", and "Entre el año de la Cuenta". At the bottom, there are four buttons: "Nuevo", "Insertar", "Actualizar", and "Eliminar".

Anexo 6.4: Caso de Uso Gestionar Activos A Largo Plazo

Gestionar Activos a Largo Plazo

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SU...	7854.0	126	2015
INMUEBLES PARA...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUS...	7854.0	245	2014

EFFECTOS POR COBRAR A LARGO PLAZO

Entre el Saldo de la Cuenta

Entre el Nomenclador de la Cuenta

Entre el año de la Cuenta

Nuevo Insertar Actualizar Eliminar

Anexo 6.5: Caso de Uso Gestionar Activos Fijos

Gestionar Activos Fijos

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SU...	7854.0	126	2015
INMUEBLES PARA ...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUST...	7854.0	245	2014

ACTIVOS FIJOS TANGIBLES

Entre el Saldo de la Cuenta

Entre el Nomenclador de la ...

Entre el año de la Cuenta

Nuevo Insertar Actualizar Eliminar

Anexo 6.6: Caso de Uso Gestionar Activos Diferidos

Gestionar Activos Diferidos

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SU...	7854.0	126	2015
INMUEBLES PARA ...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUST...	7854.0	245	2014

GASTOS DIFERIDOS A CORTO PLAZO

Entre el Saldo de la Cuenta

Entre el Nomenclador de la ...

Entre el año de la Cuenta

Anexo 6.7: Caso de Uso Gestionar Otros Activos

Gestionar Otros Activos

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SU...	7854.0	126	2015
INMUEBLES PARA...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUS...	7854.0	245	2014

CUENTAS POR COBRAR DIVERSAS -OPER...

Entre el Saldo de la Cuenta

Entre el Nomenclador de l...

Entre el año de la Cuenta

Anexo 6.8: Caso de Uso Gestionar Cuentas Reguladoras de Activos

Gestionar Cuentas Reguladoras de Activos

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SU...	7854.0	126	2015
INMUEBLES PARA...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUS...	7854.0	245	2014

EFFECTOS POR COBRAR DESCONTADOS

Entre el Saldo de la Cuenta

Entre el Nomenclador de la Cuenta

Entre el año de la Cuenta

 Nuevo  Insertar  Actualizar  Eliminar

Anexo 6.9: Caso de Uso Gestionar Pasivos Circulantes

Gestionar Pasivos Circulantes

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SU...	7854.0	126	2015
INMUEBLES PARA...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUS...	7854.0	245	2014

SOBREGIRO BANCARIO

Entre el Saldo de la Cuenta

Entre el Nomenclador de la...

Entre el año de la Cuenta

 Nuevo  Insertar  Actualizar  Eliminar

Anexo 6.10: Caso de Uso Gestionar Pasivos A Largo Plazo

Gestionar Pasivos a Largo Plazo

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SU...	7854.0	126	2015
INMUEBLES PARA...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUS...	7854.0	245	2014

EFFECTOS POR PAGAR A LARGO PLAZO

Entre el Saldo de la Cuenta

Entre el Nomenclador de la Cuenta

Entre el año de la Cuenta

Anexo 6.11: Caso de Uso Gestionar Pasivos Diferidos

Gestionar Pasivos Diferidos

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SU...	7854.0	126	2015
INMUEBLES PARA...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUS...	7854.0	245	2014

INGRESOS DIFERIDOS

Entre el Saldo de la Cuenta

Entre el Nomenclador de la Cuenta

Entre el año de la Cuenta

Anexo 6.12: Caso de Uso Gestionar Otros Pasivos

Gestionar Cuentas Otros Pasivos

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SU...	7854.0	126	2015
INMUEBLES PARA...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS AJUS...	7854.0	245	2014

Cuentas por pagar diversas

Entre el Saldo de la Cuenta

Entre el Nomenclador de la Cuenta

Entre el año de la Cuenta

Anexo 6.13: Caso de Uso Gestionar Patrimonio Neto o Capital Contable

Gestionar Cuentas del Patrimonio Neto o Capital Contable

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SU...	7854.0	126	2015
INMUEBLES PARA...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUS...	7854.0	245	2014

Entre el Saldo de la Cuenta

Entre el Nomenclador de la Cuenta

Entre el año de la Cuenta

Anexo 6.14: Caso de Uso Gestionar Cuentas Nominales Deudoras(Excepto Empresas de Seguro)

Gestionar Cuentas Nominales Deudoras(Excepto Empresas de Seguro)

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SUPE...	7854.0	126	2015
INMUEBLES PARA C...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUSTIFI...	7854.0	245	2014

DEVOLUCIONES Y REBAJAS EN VENTAS

Entre el Saldo de la Cuenta

Entre el Nomenclador de la Cuenta

Entre el año de la Cuenta

Anexo 6.15: Caso de Uso Gestionar Cuentas Nominales Acreedoras (Excepto Empresas de Seguro)

Gestionar Cuentas Nominales Acreedoras(Excepto Empresas de Seguro)

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SUPER...	7854.0	126	2015
INMUEBLES PARA CO...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUSTIFI...	7854.0	245	2014

VENTAS

Entre el Saldo de la Cuenta

Entre el Nomenclador de la Cuenta

Entre el año de la Cuenta

Anexo 6.16: Caso de Uso Gestionar Cuentas Nominales Deudoras (Empresas de Seguro)

Gestionar Cuentas Nominales Deudoras(Empresas de Seguro)

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SUP...	7854.0	126	2015
INMUEBLES PARA ...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUSTI...	7854.0	245	2014

DEVOLUCIONES DE PRIMAS

Entre el Saldo de la Cuenta

Entre el Nomenclador de la Cuenta

Entre el año de la Cuenta

Anexo 6.17: Caso de Uso Gestionar Cuentas Nominales Acreedoras (Empresas de Seguro)

Gestionar Cuentas Nominales Acreedoras (Empresas Seguro)

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SUP...	7854.0	126	2015
INMUEBLES PARA ...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUSTI...	7854.0	245	2014

INGRESOS DE LA ACTMIDAD DE SEGURO...

Entre el Saldo de la Cuenta

Entre el Nomenclador de la Cuenta

Entre el año de la Cuenta

Anexo 6.18: Caso de Uso Gestionar Compras

Gestionar Compras

Nombre de Cuenta	Saldo	Nomenclador	Año
EFFECTIVO EN CAJA	58788.0	124	2015
EFFECTIVO EN CAJA	7854.0	124	2014
DERECHO DE SU...	7854.0	126	2015
INMUEBLES PARA...	4857.0	574	2014
EFFECTIVO EN CAJA	78454.0	22	2015
ANTICIPOS A JUS...	7854.0	245	2014

COMPRAS

Entre el Saldo de la Cuenta

Entre el Nomenclador de la C...

Entre el año de la Cuenta

 Nuevo  Insertar  Actualizar  Eliminar

Anexo 6.19: Caso de Uso Mostrar Reportes de las Razones de Liquidez



Asignatura: Estudio y Análisis de los Estados Financieros

Reporte **Razones de Liquidez**

	Año: 2015	Año: 2014	Varianza
Liquidez General o Solvencia	29,87	4,24	25,64
Liquidez Inmediata	28,26	1,62	26,64
Liquidez A Corto Plazo	28,26	1,62	26,64
Liquidez de Tesorería	28,26	1,62	26,64
Liquidez Media	28,26	1,62	26,64
Capital Trabajo	91.609,00	15.166,00	
Disponibilidad	28,26	1,62	26,64

Anexo 6.20: Caso de Uso Mostrar Reportes de las Razones de Solvencia



Asignatura: Estudio y Análisis de los Estados Financieros

Reporte

Razones de Solvencia

	Año: 2015	Año: 2014	Varianza
Solvencia A Largo Plazo	0,00	0,00	0,00
Solvencia Total	2,71	3,81	-1,10

Anexo 6.21: Caso de Uso Mostrar Reportes de las Razones de Solidez



Asignatura: Estudio y Análisis de los Estados Financieros

Reporte **Razones de Solidez**

	Año: 2015	Año: 2014	Varianza
Ratios de Solidez	0,37	0,26	0,11
Estabilidad de los Activos Fijos	0,00	0,00	0,00
Estabilidad Propia de los Activos Fijos	0,00	0,00	0,00

Anexo 6.22: Caso de Uso Mostrar Reportes de las Razones de Endeudamiento



Asignatura: Estudio y Análisis de los Estados Financieros

Reporte

Razones de Endeudamiento

	Año: 2015	Año: 2014	Varianza
Razón de Endeudamiento	0,37	0,26	0,11
Endeudamiento Total	0,00	0,00	0,00
Grado de Endeudamiento	0,37	0,26	0,11
Calidad de la Deuda	0,09	0,90	-0,81

Anexo 6.23: Caso de Uso Mostrar Reportes de las Razones de Rotación



Asignatura: Estudio y Análisis de los Estados Financieros

Reporte

Razones de Rotación

	Año: 2015	Año: 2014	Varianza
Índice de Rotación de la Empresa	0,00	0,00	0,00
Rotación de los Activos Circulantes	0,00	0,00	0,00
Rotación de los Activos Fijos	0,00	0,00	0,00
Rotación del Capital de Trabajo	0,00	0,00	0,00
Ciclo de Cobro	0,00	0,00	0,00
Ciclo Pago	0.0	0.0	0.0

Anexo 6.24: Caso de Uso Mostrar Reportes de las Razones de Rentabilidad



Asignatura: Estudio y Análisis de los Estados Financieros

Reporte **Razones de Rentabilidad**

	Año: 2015	Año: 2014	Varianza
Indice del Margen Neto o de Rentabilidad de los Ingresos	0,00	0,00	0,00
Rentabilidad Económica	0,00	0,00	0,00
Rentabilidad Financiera	0,00	0,00	0,00
Apalancamiento Financiero	0,00	0,00	0,00
Composición del Índice de Rentabilidad Económica	0,00	0,00	0,00
Composición del Índice de Rentabilidad Financiera	0,00	0,00	0,00
Rentabilidad Económica Neta	0,00	0,00	0,00

Anexo 6.25: Caso de Uso Mostrar Reportes de las Razones de Ventas



Asignatura: Estudio y Análisis de los Estados Financieros

Reporte **Razones de Ventas**

	Año: 2015	Año: 2014	Varianza
Productividad	0,00	0,00	0,00