

Propuesta del módulo de configuración del “software contable para uso educativo en el programa de contaduría pública de la Universidad del Cauca”



Monografía para optar al título de ingeniero de sistemas

Modalidad: Práctica Profesional

Santiago Nieto Guaca

Asesor de la empresa: Mauro Andrés Sánchez Muñoz

Director: PhD. Wilson Libardo Pantoja Yepez

Co-Director: PhD. Julio Ariel Hurtado Alegria

Universidad del Cauca

Facultad de Ingeniería Electrónica y Telecomunicaciones

Departamento de Sistemas

Línea de investigación: Ingeniería de Software

Popayán, Noviembre de 2025

Contenido

Lista de figuras	iv
Lista de tablas	v
Capítulo 1. Introducción	1
1.1. Justificación y planteamiento del problema	1
1.2. Objetivos	3
1.2.1. Objetivo General	3
1.2.2. Objetivos Específicos	3
1.3. Metodología de trabajo	3
1.3.1. Trabajo Freelance	4
1.3.2. Kanban	5
1.3.3. Desarrollo iterativo e incremental	6
1.3.4. Iteraciones	6
Capítulo 2. Marco Teórico	11
2.1. Conceptos fundamentales	12
2.1.1. Configuración	12
2.1.2. Framework	13
2.1.3. Modelo C4	13
2.1.4. Pruebas de software	13
2.2. Tecnologías para el desarrollo	14
2.2.1. Tecnologías Frontend	14
2.2.2. Tecnologías Backend	15
2.2.3. Tecnologías para Base de Datos	16
2.3. Lineamientos división TIC	16
2.3.1. Lineamientos de diseño	16
2.3.2. Lineamientos de desarrollo	18
2.3.3. Lineamientos de usabilidad	18
Capítulo 3. Descripción de los submódulos a implementar	20
3.1. Gestión de catálogo de cuentas	21
3.2. Gestión de tarifas de impuestos	22

3.3. Gestión de terceros	23
3.4. Gestión de inventario de productos	24
3.5. Gestión de métodos de pago	25
3.6. Gestión de tipos de documentos	26
3.7. Gestión de banco y cuentas bancarias	27
3.8. Gestión de centros de costo	28
3.9. Gestión de centro de ayuda	29
3.10. Gestión de calendario contable	30
Capítulo 4. Ingeniería de requisitos	31
4.1. Actividades desarrolladas para la ingeniería de requisitos	32
4.1.1. Reuniones con los interesados	32
4.1.2. Construcción de prototipos	32
4.1.3. Definición de historias de usuario	32
4.1.4. Validación de prototipos	33
4.1.5. Validación de historias de usuario	33
4.2. Identificación de stakeholders	33
4.3. Prototipos	34
4.3.1. Vista de catálogo de cuentas	35
4.3.2. Vista de inventario de productos	36
4.3.3. Vista de terceros	37
4.3.4. Vista de impuestos	38
4.4. Requisitos funcionales	38
Capítulo 5. Arquitectura del sistema	41
Capítulo 6. Patrones de diseño	42
Capítulo 7. Pruebas realizadas a la aplicación web	43
Capítulo 8. Conclusiones	44
A. Anexo: Nombrar el anexo A de acuerdo con su contenido	45
B. Anexo: Nombrar el anexo B de acuerdo con su contenido	46
C. Anexo: Nombrar el anexo C de acuerdo con su contenido	47
Bibliografía	48

Lista de Figuras

2-1. Color institucional primario	17
2-2. Color institucional secundario	17
2-3. Otros colores destacados	17
3-1. Submódulos que componen el módulo de configuración	20
3-2. Relación del submódulo gestión de cuentas contables.	21
3-3. Relación del submódulo gestión de tarifas de impuestos.	22
3-4. Relación del submódulo gestión de terceros.	23
3-5. Relación del submódulo gestión de inventario de productos.	24
3-6. Relación del submódulo gestión de métodos de pago.	25
3-7. Relación del submódulo gestión de tipos de documentos.	26
3-8. Relación del submódulo gestión de bancos y cuentas bancarias.	27
3-9. Relación del submódulo gestión de centros de costo.	28
3-10. Relación del submódulo gestión de cuadros de ayuda.	29
3-11. Relación del submódulo gestión de calendario contable.	30
4-1. Prototipo vista catálogo de cuentas.	35
4-2. Prototipo vista inventario de productos.	36
4-3. Prototipo vista de terceros.	37
4-4. Prototipo vista de impuestos.	38

Lista de Tablas

1-1. Descripción Iteración 1	7
1-2. Descripción Iteración 2	8
1-3. Descripción Iteración 3	8
1-4. Descripción Iteración 4	9
1-5. Descripción Iteración 5	9
1-6. Descripción Iteración 6	10
2-1. Tecnologías para proyectos software de la división TIC	18
4-1. Stakeholders del proyecto	34
4-2. Lista de tareas del proyecto: historias épicas y de usuario	39

Capítulo 1. Introducción

1.1. Justificación y planteamiento del problema

La Universidad del Cauca ofrece una variedad de programas académicos orientados a la formación de profesionales en diversas áreas de conocimiento. Entre ellos, el programa de Contaduría Pública que tiene como propósito formar profesionales con una sólida capacidad en el ámbito de los negocios, responsables de garantizar la transparencia y veracidad de la información económica y financiera de las organizaciones. De igual manera, se busca formar contadores con las competencias necesarias para gestionar y administrar los recursos de empresas y entidades gubernamentales [3].

Sin embargo, se han presentado desafíos tecnológicos que impactan en la calidad de la enseñanza y el aprendizaje. Ante la carencia de un software contable propio y con el objetivo de fortalecer los procesos educativos dentro del programa de Contaduría Pública, se ha recurrido a la compra de licencias del software ERP Millenium Enterprise. No obstante, este software presenta la limitante de ser una versión académica que no da acceso a todos los módulos disponibles. Además, al ser un software externo, toda la configuración de las bases de datos debe ser gestionada por el proveedor, lo que genera retrasos y limita la capacidad de los profesores para personalizar el sistema de acuerdo a las necesidades específicas del programa.

Este tipo de dificultades subraya la importancia de integrar herramientas más eficientes y adaptadas, tal como lo sugieren varios estudios, los cuales demuestran que el uso de software en el aula apoya de manera significativa el alcance de los resultados de aprendizaje [24], al tiempo que contribuye al desarrollo y mejora de habilidades en el manejo de las Tecnologías de la Información y la Comunicación (TIC) [36]. Además, se ha evidenciado que la integración de software en los procesos de enseñanza de contabilidad ofrece beneficios tangibles, lo que permite a los estudiantes aplicar sus conocimientos en el entorno empresarial [2].

El uso del software contable en el proceso de enseñanza y aprendizaje de la contabilidad ha sido ampliamente propuesto por muchos académicos destacados [36], al punto que hoy en día se ha convertido en una necesidad su incorporación en las unidades de desarrollo académico (UDA) que promuevan el estudio y práctica del ciclo contable. Se argumenta que con la utilización de un software contable en clase se logra representar y simular las prácticas que

se llevarían en la mayoría de las organizaciones, al mismo tiempo que se ofrece un recurso pedagógico alternativo capaz de promover conocimientos relevantes y fundamentales de la contabilidad, proporcionando una mejor preparación de los estudiantes para el cambiante mundo de los negocios y la profesión contable [2].

La educación contable debe tener en cuenta los requisitos del mercado relacionados con las habilidades y conocimientos necesarios para los profesionales de la contaduría. Esto incluye ajustar el plan de estudios, métodos y herramientas utilizados en el proceso de enseñanza, incorporando el uso de tecnologías de la información (TI), lo cual fomenta la formación de profesionales con habilidades relevantes para el mercado laboral [27]. La integración de las TI fortalece dos dimensiones, en primer lugar mejorar las habilidades tecnológicas de los estudiantes y en segundo lugar, utilizar estas tecnologías como herramientas estratégicas de enseñanza que mejoran las experiencias de aprendizaje [31].

En este contexto, para que los distintos módulos del sistema contable funcionen adecuadamente, es necesario configurar y gestionar parámetros como paso inicial a la contabilidad de una empresa. Por lo tanto, se propone implementar la propuesta del módulo de configuración como parte del "Sistema Contable para Uso Educativo en el Programa de Contaduría Pública". Este módulo permitirá configurar y administrar los elementos esenciales del sistema, como cuentas contables, terceros, unidades de medida, productos, categorías, tipos de documentos, números de documentos, resolución de facturación e impuestos, entre otros. Con esta herramienta, los estudiantes podrán interactuar con un entorno contable configurable permitiendo el desarrollo de habilidades prácticas en la administración de sistemas contables, simulando escenarios empresariales.

Al integrar estas tecnologías, el programa de Contaduría Pública no solo mejorará las competencias tecnológicas de los estudiantes, sino que también reforzará su capacidad para enfrentar los desafíos del mercado laboral y aplicar sus conocimientos en contextos reales. El desarrollo del módulo de configuración implica la implementación tanto del frontend como del backend, permitiendo la integración de una interfaz intuitiva para los usuarios y una lógica de negocio robusta que permita el manejo eficiente de la información.

1.2. Objetivos

1.2.1. Objetivo General

Desarrollar la propuesta del módulo de configuración¹ del “Software contable para uso educativo en el programa de Contaduría Pública de la Universidad del Cauca”, con el propósito de ofrecer un entorno para la personalización y administración del sistema para adaptarse a las necesidades académicas y prácticas de estudiantes y docentes.

1.2.2. Objetivos Específicos

1. Definir los requisitos funcionales y no funcionales del módulo de configuración como componente del “Software contable para uso educativo en el programa de Contaduría Pública de la Universidad del Cauca”, conforme a las necesidades contables y académicas del programa.
2. Diseñar la arquitectura del módulo de configuración utilizando el modelo C4, aplicando principios de diseño modular, escalable y eficiente que permitan soportar adecuadamente los requisitos funcionales y no funcionales.
3. Implementar el módulo de configuración, abarcando tanto el front-end como el back-end, considerando los requisitos funcionales, no funcionales y la arquitectura definida.
4. Evaluar las funcionalidades implementadas del módulo de configuración, mediante pruebas de integración y aceptación, verificando su correcto funcionamiento y adecuada integración con el resto del sistema.

1.3. Metodología de trabajo

Para el desarrollo de la práctica profesional, se adoptará un enfoque de entregas iterativas e incrementales, basado en un entorno similar al trabajo freelance, caracterizado por la autonomía, la flexibilidad horaria y la gestión de proyectos de manera independiente. En este modelo, el profesional actúa de forma autónoma, ofreciendo sus servicios a sus clientes según sus necesidades específicas [29].

Sin embargo, dada la naturaleza modular del software contable a desarrollar, esta autonomía freelance se complementará con una comunicación estructurada y colaborativa con otros

¹El módulo de configuración incluye parámetros importantes del sistema que serán usados en otros módulos, como cuentas contables, terceros, inventario de productos, metodos de pago, centros de costo, bancos, tipos de documentos e impuestos.

desarrolladores involucrados en el desarrollo del mismo. Esta interacción permite una mejor integración de los diferentes módulos del sistema, requiriendo coordinación constante en aspectos como la definición de interfaces, estándares de desarrollo y protocolos de comunicación entre módulos. La modalidad freelance autónoma se adapta así a un contexto colaborativo donde, aunque cada desarrollador mantiene su independencia operativa, se establece un marco de comunicación regular que contribuye a la coherencia técnica y funcional del producto final.

Para complementar esta modalidad de trabajo híbrida, se implementará Kanban, una herramienta que permite organizar y optimizar servicios profesionales adoptando una visión holística del trabajo, enfocándose en la mejora continua desde la perspectiva del cliente. Este método permite visualizar el flujo de trabajo y la carga de tareas mediante tableros que representan las distintas etapas del proceso organizadas en columnas como “pendiente”, “en progreso”, “en prueba” y “finalizado”, lo que facilita el seguimiento del avance hasta su culminación [33].

1.3.1. Trabajo Freelance

El trabajo freelance representa una modalidad laboral donde el profesional ejerce su actividad de manera autónoma, estableciendo vínculos contractuales temporales con diversos clientes según las demandas específicas de cada proyecto; esta forma de trabajo se ha consolidado como una alternativa significativa al empleo tradicional en el panorama laboral contemporáneo, caracterizándose por su naturaleza flexible y su capacidad de adaptación a las dinámicas del mercado actual. La esencia del trabajo freelance radica en la libertad del profesional para estructurar su actividad laboral según sus propias necesidades y objetivos, sin estar sujeto a las restricciones organizacionales típicas del empleo convencional; el freelancer mantiene la autonomía para seleccionar sus proyectos, gestionar su tiempo de manera independiente, lo que le permite desarrollar una carrera profesional personalizada y diversificada. Esta modalidad laboral presenta múltiples dimensiones que la distinguen del empleo tradicional y que contribuyen a su creciente popularidad entre profesionales de diversas áreas [29]:

- **Flexibilidad Temporal y Geográfica:** Constituye uno de los aspectos más valorados de esta modalidad laboral, ya que permite al trabajador freelancer estructurar su jornada según sus preferencias personales y las exigencias específicas de cada proyecto, facilitando un equilibrio entre la vida profesional y personal. Esta flexibilidad temporal se complementa con la independencia geográfica, que elimina las limitaciones territoriales tradicionales del empleo convencional y posibilita el trabajo remoto desde cualquier ubicación.
- **Autonomía:** La autonomía se manifiesta en la capacidad del freelancer para tomar decisiones estratégicas sobre su negocio, desde la selección de clientes hasta la definición

de metodologías de trabajo, lo que fomenta el desarrollo de habilidades empresariales y la construcción de una identidad profesional sólida.

- **Diversificación de Experiencias:** La diversificación de proyectos representa otra dimensión significativa, ya que el freelancer puede involucrarse en iniciativas de diferentes sectores y naturalezas, enriqueciendo su experiencia profesional y ampliando su red de contactos, esta variedad no solo contribuye al crecimiento profesional, sino que también reduce los riesgos asociados a la dependencia de un único empleador o sector económico.

1.3.2. Kanban

El método Kanban representa un enfoque integral de gestión del trabajo que se fundamenta en la visualización y optimización de los flujos de trabajo, particularmente efectivo para servicios profesionales y trabajo del conocimiento. Este método adopta una perspectiva holística que prioriza la mejora continua de los servicios desde la óptica del cliente, permitiendo a las organizaciones desarrollar una comprensión profunda de cómo se desplaza el trabajo a través de sus procesos. La implementación de Kanban facilita una gestión empresarial más eficiente al proporcionar visibilidad sobre los riesgos asociados a la entrega de servicios, mientras que simultáneamente desarrolla la capacidad adaptativa organizacional necesaria para responder de manera ágil a las cambiantes necesidades del mercado y las expectativas de los clientes. Aunque inicialmente reconocido por su aplicación en equipos para mitigar la sobrecarga laboral y recuperar el control operativo, el verdadero potencial de Kanban se manifiesta cuando se implementa a escala organizacional, abarcando múltiples equipos y departamentos. En este amplio contexto, Kanban trasciende su función como herramienta de gestión de tareas para convertirse en un potente instrumento de desarrollo organizacional orientado al servicio, generando oportunidades significativas de mejora estructural y operativa [33].

Para materializar estos principios conceptuales, los tableros Kanban constituyen el mecanismo de visualización más ampliamente utilizado en la implementación de sistemas Kanban. Estos tableros operan bajo una lógica direccional uniforme que facilita el seguimiento del progreso del trabajo: los elementos ingresan por el lado izquierdo del tablero y avanzan hacia la derecha hasta completar su ciclo de entrega de valor al cliente. La arquitectura de un sistema Kanban establece puntos claramente definidos de compromiso y entrega. Los elementos de trabajo pueden variar considerablemente en naturaleza y magnitud, abarcando desde tareas individuales y requisitos específicos hasta proyectos completos y paquetes de productos en tableros de nivel estratégico, adaptándose así a diversos contextos organizacionales como historias de usuario en desarrollo de software, procesos de recursos humanos o desarrollo de productos. Fundamentado en el principio de “comenzar con lo que se hace actualmente”, el tablero Kanban refleja fielmente el flujo operativo real en lugar de una configuración ideal-

zada, utilizando columnas para representar las etapas del proceso y carriles para distribuir la capacidad según diferentes tipos de trabajo o proyectos [33].

1.3.3. Desarrollo iterativo e incremental

En un desarrollo iterativo e incremental el proyecto se organiza en varios bloques de tiempo conocidos como iteraciones. Las iteraciones funcionan como pequeños proyectos, en cada una se repite un flujo de trabajo similar (de ahí el término “iterativo”) para ofrecer un resultado terminado sobre el producto final, permitiendo que el cliente reciba los beneficios del proyecto de manera progresiva, cada requisito debe finalizarse dentro de una sola iteración, donde se ejecutan todas las tareas necesarias para completarlo (incluyendo pruebas y documentación) asegurando que esté listo para entregarse al cliente con el mínimo esfuerzo adicional, así se evita dejar actividades críticas relacionadas con la entrega de requisitos para el final del proyecto. Durante cada iteración el equipo mejora el producto (realizando una entrega incremental) basándose en los resultados alcanzados en iteraciones previas, incorporando nuevos objetivos o requisitos o refinando los ya completados. Un elemento importante para dirigir el desarrollo iterativo e incremental es la priorización de los objetivos o requisitos según el valor que generan para el cliente [7].

1.3.4. Iteraciones

A continuación se muestran las diferentes iteraciones con las actividades desarrolladas en cada una de ellas.

Tabla 1-1.: Descripción Iteración 1

Iteración 1	
Fase	Análisis de requisitos y planificación de tareas por prioridades
Descripción	Identificación y documentación de los requisitos funcionales y no funcionales del módulo de configuración. Priorización de funcionalidades según su valor para el interesado.
Roles involucrados	<ul style="list-style-type: none">■ Analista■ Diseñador■ Cliente
Actividades	<ul style="list-style-type: none">■ Reuniones presenciales y virtuales■ Contrucción de historias de usuario■ Construcción de prototipos
Rol del practicante	<ul style="list-style-type: none">■ Analista■ Diseñador

Tabla 1-2.: Descripción Iteración 2

Iteración 2	
Fase	Diseño de arquitectura
Descripción	Concepción de la arquitectura del sistema y lineamientos de desarrollo.
Roles involucrados	<ul style="list-style-type: none"> ■ Arquitecto. ■ Desarrollador. ■ Director de grado.
Actividades	<ul style="list-style-type: none"> ■ Revisión de la documentación. ■ Construcción de diagramas del sistema haciendo uso del modelo C4.
Rol del practicante	<ul style="list-style-type: none"> ■ Arquitecto. ■ Desarrollador.

Tabla 1-3.: Descripción Iteración 3

Iteración 3	
Fase	Desarrollo
Descripción	Desarrollo de funcionalidades priorizadas.
Roles involucrados	<ul style="list-style-type: none"> ■ Desarrollador. ■ Cliente.
Actividades	<ul style="list-style-type: none"> ■ Implementación de funcionalidades.
Rol del practicante	<ul style="list-style-type: none"> ■ Desarrollador

Tabla 1-4.: Descripción Iteración 4

Iteración 4	
Fase	Pruebas unitarias e integración
Descripción	Validación del correcto funcionamiento de las funcionalidades implementadas mediante pruebas unitarias individuales y pruebas de integración.
Roles involucrados	<ul style="list-style-type: none">■ Desarrollador■ QA
Actividades	<ul style="list-style-type: none">■ Pruebas unitarias.■ Pruebas de integración.
Rol del practicante	<ul style="list-style-type: none">■ Desarrollador■ Tester

Tabla 1-5.: Descripción Iteración 5

Iteración 5	
Fase	Pruebas de aceptación con usuarios clave
Descripción	Verificación de la usabilidad, funcionalidad y adecuación del sistema para el entorno educativo contable.
Roles involucrados	<ul style="list-style-type: none">■ Desarrollador■ QA■ Cliente
Actividades	<ul style="list-style-type: none">■ Pruebas funcionales.■ Pruebas de aceptación.
Rol del practicante	<ul style="list-style-type: none">■ Desarrollador■ Tester

Tabla 1-6.: Descripción Iteración 6

Iteración 6	
Fase	Revisión de entregas y análisis de posibles mejoras
Descripción	Evaluación integral del módulo de configuración desarrollado, analizando el cumplimiento de objetivos y requisitos establecidos. Se identifican oportunidades de mejora basadas en el feedback de usuarios y resultados de pruebas.
Roles involucrados	<ul style="list-style-type: none"> ■ Desarrollador ■ Tester ■ Stakeholders ■ Usuario final
Actividades	<ul style="list-style-type: none"> ■ Revisión de entregables y documentación ■ Análisis de feedback de usuarios ■ Identificación de mejoras potenciales
Rol del practicante	<ul style="list-style-type: none"> ■ Analista ■ Desarrollador

Capítulo 2. Marco Teórico

La enseñanza de contabilidad en el siglo XXI requiere de profesionales con conocimientos sólidos y habilidades que permitan motivar a los estudiantes. Gracias a la tecnología, se ha facilitado la creación de herramientas para el aprendizaje individualizado [20], impulsando modelos como Computer Assisted Learning (CAL) y otros basados en Tecnologías de la Información y la Comunicación (TIC) [35].

El modelo CAL es el que se ha venido implementando en la educación contable al integrar el software como parte esencial en el proceso de enseñanza y aprendizaje [19]. Este modelo permite aplicar diversas situaciones que ayudan al estudiante a comprender conceptos y generar experiencias prácticas [24], ofreciendo un estilo de aprendizaje enriquecedor que aumenta la motivación y facilita el aprendizaje activo [19]. Sin embargo, es fundamental que no solo se aprenda el uso del software, sino que este sea empleado como herramienta pedagógica [1]. Además, CAL beneficia a las universidades al optimizar costos y mejorar la calidad educativa [9], reforzando su imagen como instituciones modernas [24].

Debido a lo anterior, el dominio de las TIC se ha identificado como un factor diferenciador en contabilidad y finanzas [28]. La International Federation of Accountants (IFAC), organismo internacional que regula la profesión contable y promueve estándares de formación, enfatiza en el desarrollo de habilidades técnicas, organizacionales y sociales en los contadores [34]. Asimismo, organizaciones como CPA Australia, la American Accounting Association (AAA) y el American Institute of Certified Public Accountants (AICPA) promueven la incorporación de tecnologías emergentes en la educación contable [37]. La adopción de estas tecnologías debe formar parte de una estrategia educativa integral [32].

El concepto de contador híbrido surge como respuesta a las nuevas exigencias del mercado laboral, donde se requiere que los profesionales combinen conocimientos contables con habilidades en tecnologías de la información [28]. Esto ha impulsado a las universidades a mejorar la formación académica para responder a la demanda empresarial [32]. En este contexto, la IFAC, a través del International Accounting Education Standards Board (IAESB), ha desarrollado normativas internacionales como la International Education Standard 2 (IES 2), la cual establece competencias esenciales en TIC para contadores [6]. Estas competencias incluyen el análisis de datos, la gestión de riesgos y la optimización de sistemas organizacionales, elementos clave para la formación y desempeño profesional en contabilidad [6].

Finalmente, el rápido avance tecnológico en el ámbito contable ha llevado a una evolución en las herramientas utilizadas por los profesionales del sector. Tecnologías como la digitalización de datos, la computación en la nube, la inteligencia artificial (IA), las tecnologías financieras (FinTech), el blockchain y el análisis de datos masivos (Big Data) han transformado los procesos contables y financieros [6]. La adopción y dominio de estas herramientas no solo mejora la eficiencia de las organizaciones, sino que también fortalece la competitividad de los contadores en el mercado laboral global [15]. En este sentido, la formación académica debe evolucionar constantemente para preparar a los futuros contadores ante los desafíos de la era digital [36].

2.1. Conceptos fundamentales

A continuación se definen los conceptos y tecnologías empleadas durante el desarrollo de la práctica profesional, con el propósito de establecer un marco de referencia común para la comprensión del presente documento.

2.1.1. Configuración

En contabilidad, la configuración de un software contable se refiere a la implementación y adaptación de los ajustes, procesos, datos y reglas de negocio dentro del sistema para que se alineen precisamente con las necesidades operativas y fiscales de una empresa [14]. Esto implica tomar una herramienta estándar y personalizar sus módulos y funcionalidades, como la gestión de facturación, la contabilidad, el manejo de almacén y la configuración de datos fiscales o bancarios, para reflejar los flujos de trabajo y los requerimientos específicos del negocio [14].

Una configuración adecuada permite gestionar las finanzas y operaciones de manera eficiente, ofreciendo simplicidad y control, similar a cómo un motor bien ajustado impulsa un automóvil [4]. Sin una estrategia clara y una configuración correcta, las empresas corren el riesgo de sufrir interrupciones costosas, ineficiencias operativas y no alcanzar los objetivos de la implementación del sistema [12].

Además, permite centralizar la información y obtener una visión global en tiempo real, automatizar procesos, mejorar la toma de decisiones, facilitar el cumplimiento normativo y definir roles de usuario y entornos de trabajo [18]. Es vital para gestionar procesos, procedimientos y proteger la información, asegurando que las transacciones se registren, administren y reporten de manera precisa dentro del contexto específico de la empresa [11].

2.1.2. Framework

Un framework es una colección integrada de herramientas, aplicaciones y librerías que implementa estándares y patrones de diseño establecidos. Su objetivo principal es proporcionar una estructura cohesiva que facilite el desarrollo de software para fines específicos, permitiendo que los desarrolladores se enfoquen en la lógica de negocio fundamental sin preocuparse por tareas repetitivas o por la implementación de aspectos técnicos complejos y especializados [13].

2.1.3. Modelo C4

El Modelo C4 está estructurado mediante niveles de abstracción organizados jerárquicamente, abarcando desde sistemas de software hasta contenedores, componentes y código fuente. Esta arquitectura se visualiza a través de diagramas dispuestos en diferentes niveles que muestran el contexto del sistema, la organización de contenedores, la estructura de componentes y los detalles del código. Una ventaja significativa de este modelo es que no depende de ninguna notación particular ni requiere herramientas específicas para su aplicación práctica [17].

2.1.4. Pruebas de software

Las pruebas de software constituyen un proceso sistemático destinado a evaluar y confirmar que una aplicación o sistema funciona correctamente según las especificaciones establecidas. Sus principales ventajas incluyen la detección temprana de fallos y el incremento del rendimiento general del sistema. Las pruebas resultan más eficientes cuando se ejecutan de manera continua, integrándose desde las fases iniciales de diseño y manteniéndose activas durante todo el ciclo de desarrollo hasta llegar al entorno de producción. Este enfoque continuo elimina la necesidad de esperar a que el producto esté completamente terminado para iniciar las validaciones. Existen diferentes tipos pruebas de software, algunas de ellas son [21]:

- **Pruebas de aceptación:** Verifican si todo el sistema funciona según lo previsto.
- **Pruebas de integración:** Aseguran que los diferentes elementos o funcionalidades del software operan de manera coordinada e integrada.
- **Pruebas unitarias:** Validan que cada módulo individual del software opera conforme a lo previsto. Un módulo representa la parte más pequeña y verificable de una aplicación.
- **Pruebas funcionales:** Verifican las funcionalidades del sistema simulando situaciones empresariales reales basadas en los requerimientos funcionales establecidos. Las pruebas

de caja negra constituyen un método común para validar estas funcionalidades.

- **Pruebas de rendimiento:** Evalúan el comportamiento del software bajo diversas condiciones de carga operativa. Las pruebas de carga, por ejemplo, se emplean para medir el desempeño del sistema en escenarios de uso real.
- **Pruebas de usabilidad:** Validan la facilidad con que un usuario puede interactuar con un sistema o aplicación web para llevar a cabo una tarea específica.

2.2. Tecnologías para el desarrollo

Para el desarrollo del módulo es necesario utilizar diversas tecnologías que permitan su correcto funcionamiento, permitiendo tanto la interacción con los usuarios como el procesamiento de la información. Estas herramientas han sido seleccionadas con el objetivo de mejorar la experiencia del usuario, la eficiencia en el manejo de datos y la estabilidad del sistema. A continuación, se presentan las principales tecnologías que serán empleadas, abarcando desde el diseño de la interfaz hasta la gestión de la información en el servidor y la base de datos.

2.2.1. Tecnologías Frontend

Angular y Tailwind: En el desarrollo frontend, la combinación de herramientas que atienden tanto la lógica funcional como el diseño visual de las aplicaciones resulta fundamental para construir soluciones eficientes, escalables y mantenibles. En este contexto, Angular y Tailwind CSS se presentan como tecnologías que, aunque abordan áreas distintas, pueden integrarse de forma complementaria para optimizar el proceso de desarrollo.

Angular es un framework que proporciona una arquitectura robusta basada en componentes, permitiendo estructurar las aplicaciones en vistas jerárquicas que interactúan entre sí. Estas vistas se comunican mediante la inyección de dependencias con servicios que encapsulan la lógica de negocio o funcionalidades reutilizables, sin necesidad de estar directamente ligados a la interfaz gráfica. Además de su enfoque estructural, Angular incorpora herramientas integradas para la realización de pruebas unitarias y de integración, utilizando frameworks como Jasmine y Karma, lo cual favorece el desarrollo confiable y orientado a pruebas desde las primeras fases del proyecto [30].

Por su parte, Tailwind CSS es un framework de estilos que permite aplicar directamente clases predefinidas en el HTML, eliminando la necesidad de escribir archivos CSS personalizados. Este enfoque promueve la rapidez en la maquetación de interfaces, permitiendo ajustes visuales inmediatos sin salir del flujo de trabajo del código. Tailwind favorece un diseño consistente, reutilizable y altamente personalizable, facilitando la implementación de

interfaces modernas y adaptables a distintas resoluciones y dispositivos [23].

2.2.2. Tecnologías Backend

El desarrollo backend moderno se fundamenta en la integración de diversas tecnologías que, en conjunto, permiten estructurar aplicaciones web eficientes, escalables y orientadas a servicios. Tecnologías como JSON, REST, Spring Framework, Java y herramientas de registro de servicios como Eureka no solo cumplen funciones individuales dentro del ecosistema de desarrollo, sino que trabajan de forma articulada para facilitar el procesamiento, la entrega y la organización de los datos en una aplicación.

En este flujo de trabajo, JSON actúa como el formato de intercambio de datos por excelencia, gracias a su ligereza, legibilidad y compatibilidad con múltiples lenguajes. Su estructura basada en pares clave-valor lo hace ideal para representar objetos y transmitir información entre cliente y servidor de forma sencilla y estructurada [22]. Este formato cobra especial relevancia dentro de arquitecturas REST, donde los datos se transportan principalmente en formato JSON. REST define un estilo de arquitectura que permite que los distintos sistemas se comuniquen a través de HTTP mediante operaciones estándar (GET, POST, PUT, DELETE), y que los recursos se identifiquen a través de URLs claras [25]. Al combinar REST con JSON, se obtiene una interfaz de comunicación eficiente y estandarizada para consumir y exponer servicios web.

Para implementar esta lógica de negocio entra en juego Spring Framework, una herramienta robusta que proporciona todos los componentes necesarios para desarrollar APIs de manera modular y mantenible. Spring no solo ofrece soporte para la construcción de controladores que gestionen solicitudes HTTP y devuelvan respuestas en formato JSON, sino que también simplifica aspectos internos como la inyección de dependencias, la gestión de datos o la configuración de pruebas [26]. Su compatibilidad nativa con REST y su estructura basada en componentes lo convierten en un puente ideal entre los principios arquitectónicos REST y el desarrollo práctico de servicios web.

En arquitecturas distribuidas de microservicios, se hace necesario incorporar herramientas especializadas para el registro y descubrimiento de servicios. Eureka Server, desarrollado inicialmente por Netflix, cumple esta función al actuar como un servidor centralizado donde los microservicios pueden registrarse automáticamente para ser localizados por otros servicios. Esta herramienta elimina la necesidad de definir manualmente las direcciones de cada servicio, gestionando dinámicamente el proceso de descubrimiento y comunicación entre componentes distribuidos [16].

Todo este ecosistema se construye sobre Java, el lenguaje base que aporta características como portabilidad, orientación a objetos, robustez y seguridad. Java no solo permite implementar la lógica del negocio y manipular estructuras complejas de datos, sino que también se

adapta perfectamente al uso de frameworks como Spring, asegurando rendimiento y fiabilidad en aplicaciones backend de cualquier escala [38].

2.2.3. Tecnologías para Base de Datos

La gestión eficiente de la persistencia de datos en una aplicación requiere herramientas que reduzcan la complejidad del acceso a bases de datos relacionales y que, al mismo tiempo, se integren de manera fluida con el lenguaje de programación utilizado. En este sentido, Java Persistence API (JPA) e Hibernate forman una combinación poderosa que facilita el trabajo con bases de datos desde entornos orientados a objetos como Java.

JPA proporciona una especificación estándar para mapear clases Java a tablas de una base de datos, permitiendo manipular datos como objetos sin tener que interactuar directamente con el lenguaje SQL. Esto se traduce en una capa de abstracción que permite al desarrollador realizar operaciones como insertar, actualizar, eliminar o consultar registros, utilizando una sintaxis más cercana al paradigma orientado a objetos [39].

Por su parte, Hibernate es una implementación concreta y extendida de esa especificación. No solo cumple con los lineamientos definidos por JPA, sino que añade funcionalidades avanzadas que automatizan procesos comunes como la sincronización entre objetos y registros, el control de transacciones o el uso de caché para mejorar el rendimiento. Hibernate actúa como el motor que lleva a cabo las operaciones indicadas por JPA, reduciendo la necesidad de código repetitivo y facilitando el mantenimiento del sistema [5].

2.3. Lineamientos división TIC

2.3.1. Lineamientos de diseño

El documento “Lineamientos de diseño UX/UI para proyectos de desarrollo de software en la División TIC de la Universidad del Cauca”, desarrollado por el área de soporte y desarrollo en su primera versión, define directrices para el diseño en proyectos de desarrollo de software. Este incluye consideraciones sobre tipografía, esquemas cromáticos y enfoques de diseño, ya que los “proyectos deben mantener la identidad visual del Alma Mater, con la finalidad de reforzar la institucionalidad y el posicionamiento de la marca en diversos entornos digitales” [8]. Entre estas directrices se encuentran especificaciones tipográficas que determinan el uso de las fuentes Titillium Web y Open Sans, así como definiciones cromáticas que establecen diferentes gamas de colores y su aplicación. En las figuras 2-1, 2-2 y 2-3 se presentan algunos de los colores fundamentales.

Primario	Light	Dark
R: 0 G: 0 B: 112	R: 80 G: 86 B: 172	R: 7 G: 24 B: 74
#000066	#5056AC	#07184A

Figura 2-1.: Color institucional primario

Secundario	Light	Dark
R: 157 G: 3 B: 12	R: 219 G: 20 B: 28	R: 105 G: 0 B: 7
#9D0311	#DB141C	#690007

Figura 2-2.: Color institucional secundario

Error	Terciario	Seguridad
R: 255 G: 109 B: 10	R: 29 G: 114 B: 211	R: 224 G: 224 B: 255
#FF6D0A	#1D72D3	#E0E0FF

Figura 2-3.: Otros colores destacados

2.3.2. Lineamientos de desarrollo

La Universidad del Cauca, a través de su división TIC, establece directrices de desarrollo que determinan, entre otros aspectos, las tecnologías a utilizar para el desarrollo de aplicaciones institucionales, las cuales incluyen las siguientes [10]:

Tabla 2-1.: Tecnologías para proyectos software de la división TIC

Tipo	Tecnología
Bases de datos SQL	<ul style="list-style-type: none"> ▪ Oracle. ▪ MySQL. ▪ PostgreSQL.
Bases de datos no SQL	<ul style="list-style-type: none"> ▪ MongoDB. ▪ Cassandra.
Bases de datos Real Time	<ul style="list-style-type: none"> ▪ Firebase
Back-end	<ul style="list-style-type: none"> ▪ Spring Boot ▪ Java
Front-end	<ul style="list-style-type: none"> ▪ JavaScript ▪ Vue ▪ React con Next ▪ Angular con TypeScript

2.3.3. Lineamientos de usabilidad

En el documento “Lineamientos normativos y pautas de usabilidad para el diseño de interfaces en los proyectos de desarrollo software en la División TIC de la Universidad del Cauca” se definen criterios de usabilidad, entre los cuales se encuentran [9]:

- **Tipografía legible:** Uso de fuentes con dimensiones y contraste apropiados que garanticen una lectura cómoda para los usuarios.
- **Colores accesibles:** Implementación de esquemas de color que mantengan la identidad visual institucional, asegurando un contraste óptimo entre el texto y el fondo para promover la legibilidad y reducir el cansancio visual.
- **Navegación intuitiva:** Desarrollo de una arquitectura gráfica transparente y de fácil comprensión mediante el uso de encabezados y divisiones apropiadas que orienten al usuario en la búsqueda de información.
- **Contenido accesible:** Presentación de contenido claro y directo, empleando un lenguaje accesible y evitando terminología técnica innecesaria.

Capítulo 3. Descripción de los submódulos a implementar

Este capítulo describe el módulo de configuración mediante la descomposición en submódulos funcionales, presentados en un lenguaje de alto nivel para facilitar la comprensión de cualquier lector, los cuales permiten la operatividad, personalización y adaptabilidad del sistema a contextos académicos. Se detallan las responsabilidades específicas de cada submódulo y cómo estos influyen en los demás módulos del sistema contable educativo, proporcionando un concepto general de cómo el sistema gira en torno a los parámetros establecidos en la configuración, las relaciones técnicas se detallarán en capítulos posteriores. Estos submódulos cubren gran parte de las dimensiones necesarias para simular entornos empresariales, desde la gestión de cuentas contables e impuestos hasta el control de inventarios, métodos de pago, centros de costo y otros parámetros operativos que se detallarán a continuación.

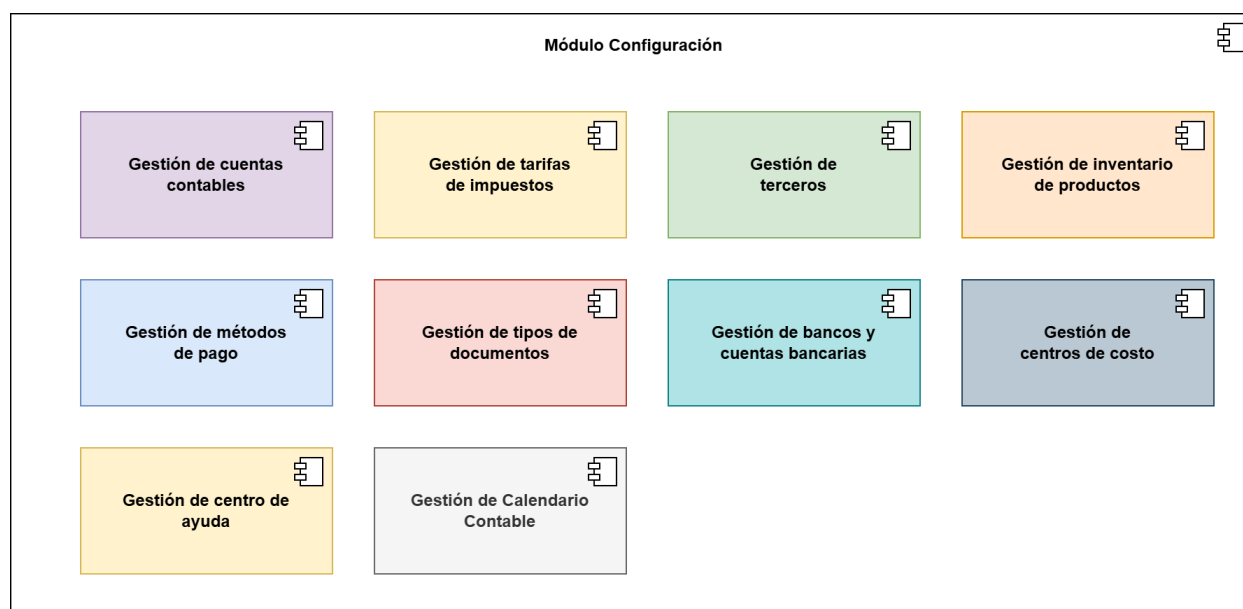


Figura 3-1.: Submódulos que componen el módulo de configuración

El módulo de configuración es fundamental para el funcionamiento del sistema contable, ya que permite a los administradores establecer y mantener los datos maestros y parámetros que

rigen las operaciones de todos los demás módulos. A continuación, se detallan los submódulos identificados.

3.1. Gestión de catálogo de cuentas

Tiene como objetivo mantener una estructura contable actualizada y adaptada a las necesidades del sistema, proporcionando funcionalidades esenciales para la creación, modificación, eliminación, importación desde archivos planos, exportación a archivos planos y consulta de cuentas contables, lo que permite una administración integral del catálogo de cuentas.

Este submódulo constituye el núcleo fundamental del sistema contable, al establecer relaciones críticas con los módulos de Inventario (Promedio Ponderado y PEPS) para la generación automática de asientos contables relacionados con el manejo de inventarios; conectarse con el Módulo Comercial para el registro de transacciones de compras y ventas; interactuar con los módulos de Tesorería y Cartera para el registro de movimientos financieros; servir como base para los módulos Contable-Comercial y Contable Cartera, que dependen directamente del catálogo para generar asientos contables; proporcionar la información necesaria para los reportes financieros, tanto de Libros Auxiliares como de Estados Financieros y alimentar el Módulo Educativo Contable, que utiliza esta información para ejercicios de simulación. De esta manera, se consolida como el elemento vertebral que garantiza la coherencia y funcionalidad de todo el ecosistema contable. A continuación se muestran de manera gráfica las relaciones correspondientes del submódulo de Gestión de Cuentas Contables.

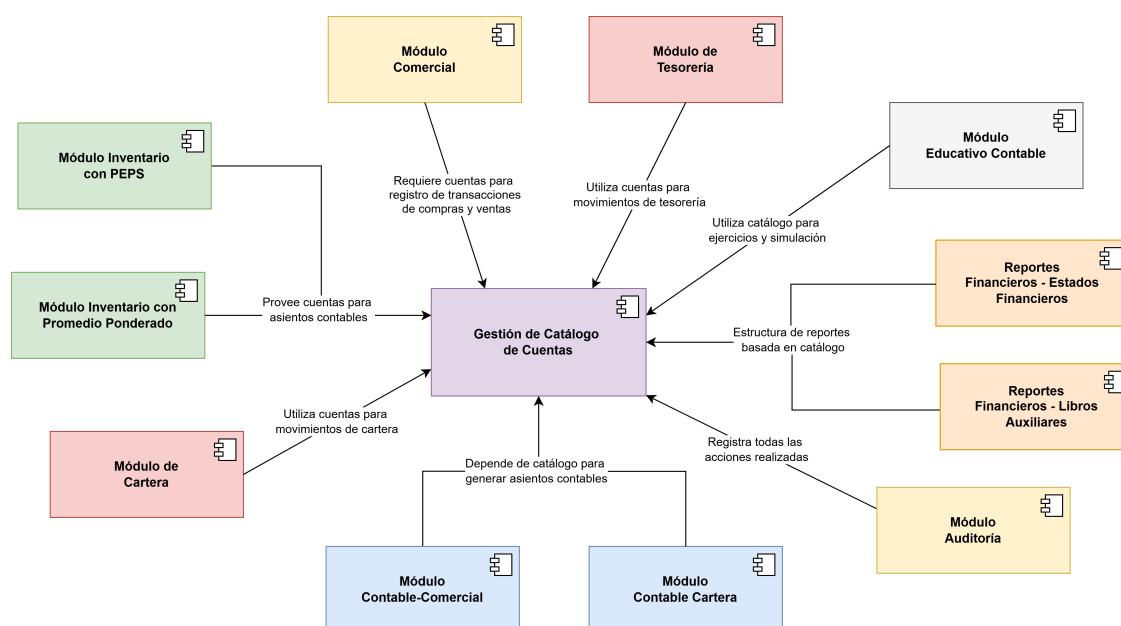


Figura 3-2.: Relación del submódulo gestión de cuentas contables.

3.2. Gestión de tarifas de impuestos

Tiene como objetivo establecer los valores de las tarifas de impuestos asociadas a las transacciones del sistema, ofreciendo funcionalidades que incluyen la creación, modificación, eliminación y consulta de tarifas de impuestos, así como la asociación de cuentas contables auxiliares a cada tarifa correspondiente. Este submódulo se conecta con los módulos de Inventario (Promedio Ponderado y PEPS), donde las tarifas de impuestos pueden influir en el costo de los productos en inventario; resulta esencial para el Módulo Comercial, al facilitar el cálculo de impuestos en facturas de compra y venta y se relaciona con el Módulo Contable-Comercial, donde los asientos contables generados por transacciones comerciales incorporan los impuestos calculados conforme a las tarifas establecidas. De esta forma, constituye un elemento central para el manejo tributario en todas las operaciones del sistema. A continuación se muestran de manera gráfica las relaciones correspondientes del submódulo de Gestión de tarifas de impuestos.

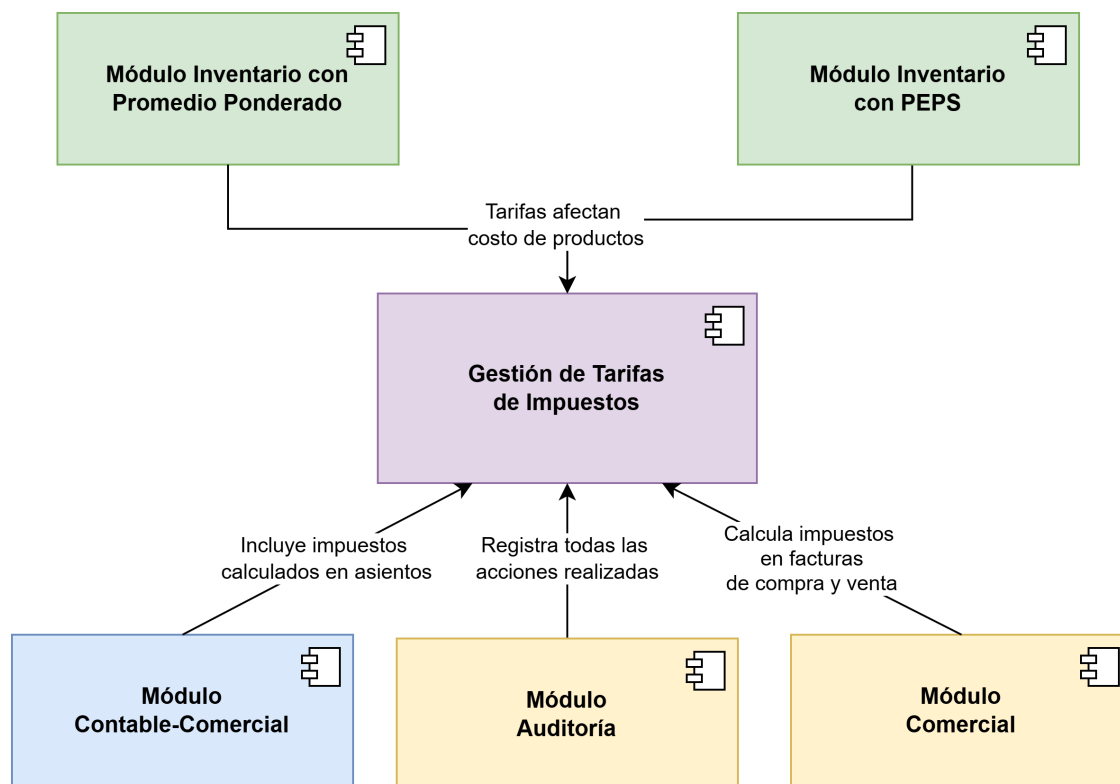


Figura 3-3.: Relación del submódulo gestión de tarifas de impuestos.

3.3. Gestión de terceros

Tiene como objetivo mantener una base de datos actualizada de clientes, proveedores y otras entidades relevantes para las operaciones del sistema, proporcionando funcionalidades que incluyen la creación manual de registros, la creación automática desde archivos PDF (RUT), la modificación, la importación desde archivos planos, la exportación a archivos planos, la consulta y la visualización completa de terceros. Este submódulo se relaciona con el Módulo Comercial, donde resulta esencial para identificar clientes en operaciones de venta y proveedores en procesos de compra; se vincula con el Módulo de Tesorería, permitiendo la gestión de pagos a proveedores; se relaciona con el Módulo de Cartera, posibilitando la gestión de cobros a clientes; interactúa con los Reportes Financieros (tanto de Libros Auxiliares como de Estados Financieros), donde los reportes pueden filtrarse y presentarse por tercero específico y se vincula con el Módulo Educativo Contable, que puede utilizar información de terceros para desarrollar simulaciones y ejercicios prácticos. De esta manera, constituye la base informativa que soporta las relaciones comerciales y financieras del sistema. A continuación se muestran de manera gráfica las relaciones correspondientes del submódulo de Gestión de terceros.

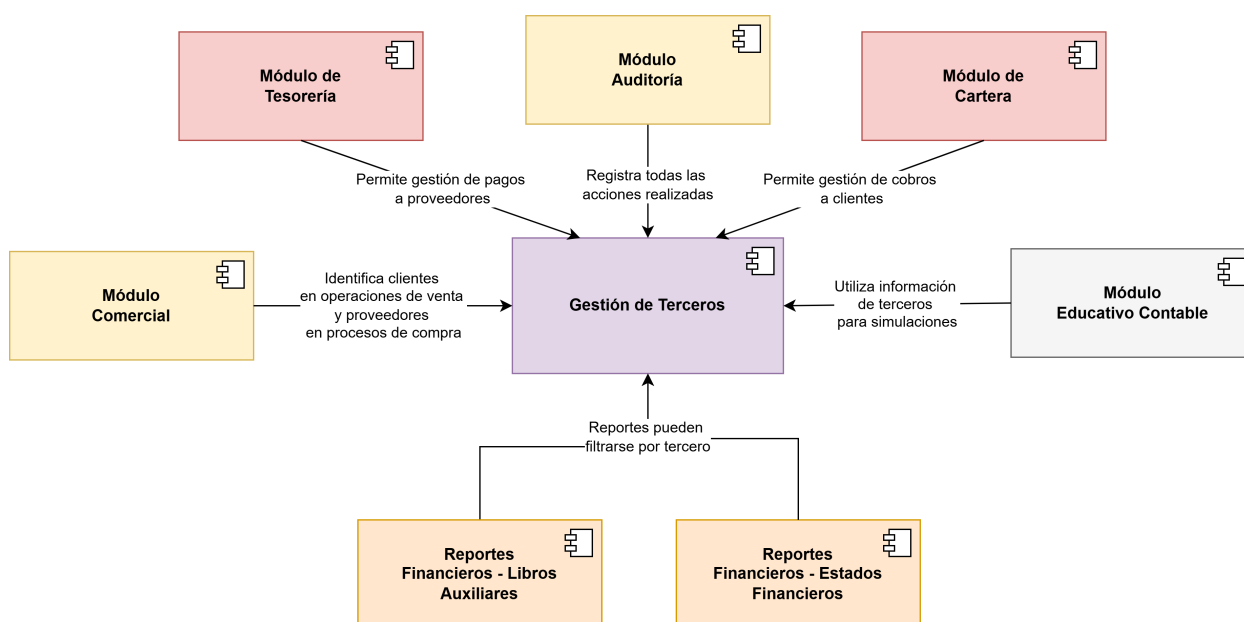


Figura 3-4.: Relación del submódulo gestión de terceros.

3.4. Gestión de inventario de productos

Tiene como objetivo mantener el catálogo de productos actualizado y correctamente asociado a las cuentas contables e impuestos, permitiendo su gestión completa a través de funcionalidades que incluyen la creación, modificación, eliminación, importación desde archivo plano, exportación a archivo plano y consulta de productos en el inventario. Provee información detallada de los productos a los módulos de Inventario con Promedio Ponderado e Inventario con PEPS para la gestión de inventarios correspondiente, constituye el elemento central del Módulo Comercial donde los productos representan el núcleo de las transacciones de compra y venta, y se relaciona con el Módulo Contable-Comercial donde los movimientos de productos afectan directamente el costo de ventas y la generación de asientos contables, estableciéndose así como el repositorio fundamental que contiene toda la información necesaria para el manejo comercial y contable de los productos dentro del sistema. A continuación se muestran de manera gráfica las relaciones correspondientes del submódulo de Gestión de inventario de productos.

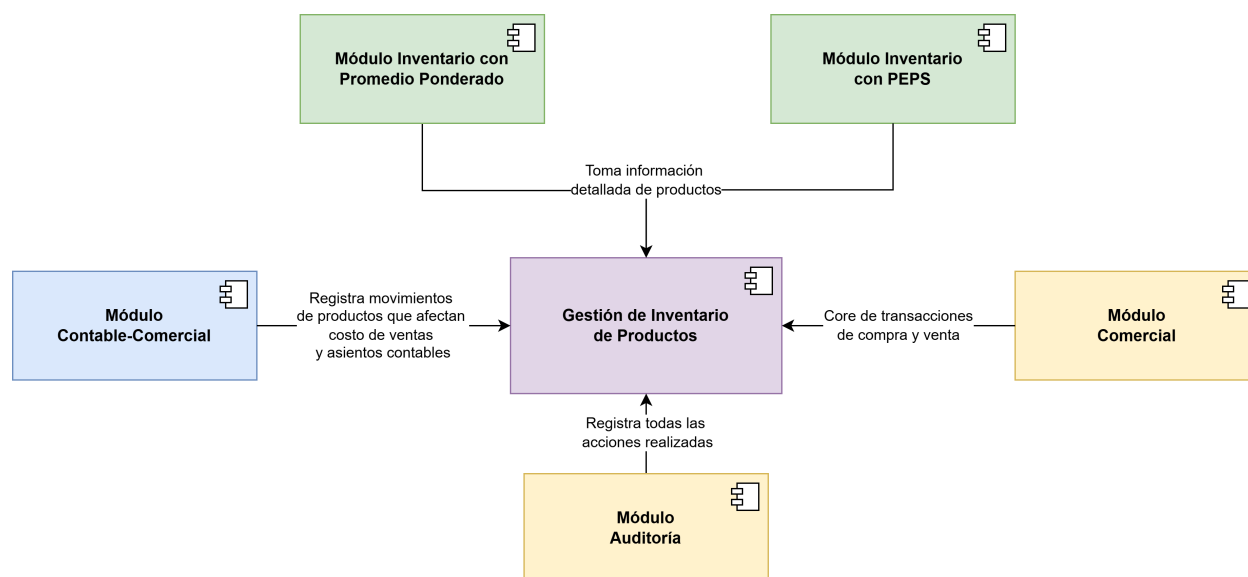


Figura 3-5.: Relación del submódulo gestión de inventario de productos.

3.5. Gestión de métodos de pago

Tiene como objetivo establecer los métodos de pago que se usarán como información adicional en los registros contables de comprobantes de egreso y recibos de caja, estableciendo la correcta equivalencia contable de cada método de pago a través de funcionalidades que incluyen la creación, modificación y eliminación de métodos de pago. Este submódulo se conecta con el Módulo de Tesorería, permitiendo la selección del método de pago para los comprobantes de egreso; se integra con el Módulo de Cartera, posibilitando la selección del método de pago para los recibos de caja y se relaciona con los módulos Contable-Comercial y Contable Cartera, donde los asientos contables generados por los pagos y cobros se basan en la parametrización establecida para cada método de pago. De esta forma, constituye el elemento que define y controla la aplicación contable de las diferentes formas de pago utilizadas en las transacciones del sistema contable. A continuación se muestran de manera gráfica las relaciones correspondientes del submódulo de Gestión de métodos de pago.

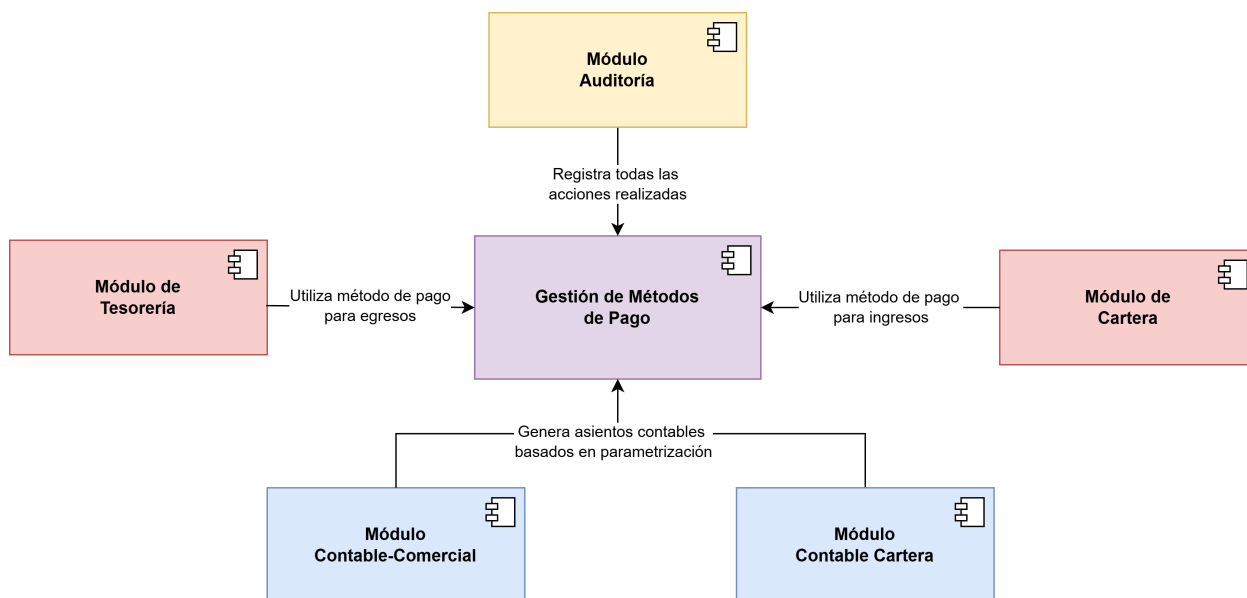


Figura 3-6.: Relación del submódulo gestión de métodos de pago.

3.6. Gestión de tipos de documentos

Su objetivo es clasificar y configurar los diferentes tipos de documentos que se manejan en el sistema (factura de venta, factura de compra, recibo de caja, comprobante de egreso, nota de crédito, nota de débito), estableciendo una adecuada clasificación, flujo y registro contable de las operaciones a través de funcionalidades que incluyen la creación, modificación y eliminación de tipos de documentos, así como la definición de prefijos para cada tipo de documento. Este submódulo se conecta con el Módulo Comercial, definiendo los tipos de documentos comerciales como facturas, cotizaciones y órdenes; se integra con los módulos de Tesorería y Cartera, estableciendo los tipos de documentos de tesorería (comprobantes de egreso) y cartera (recibos de caja); se relaciona con los módulos Contable-Comercial y Contable Cartera, donde la generación de asientos contables está ligada al tipo de documento y se vincula con el Módulo de Auditoría, que filtra por tipos de documentos para visualizar los registros de acciones realizadas en el sistema. De esta forma, constituye el elemento normativo que define y controla la estructura documental de las operaciones. A continuación se muestran de manera gráfica las relaciones correspondientes del submódulo de Gestión de tipos de documentos.

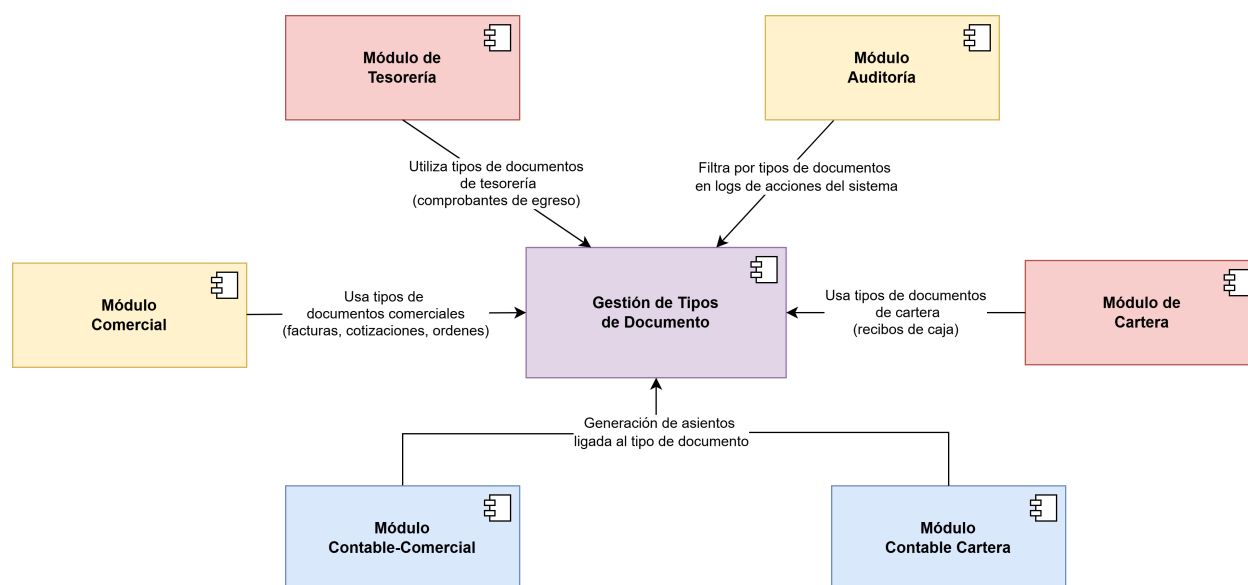


Figura 3-7.: Relación del submódulo gestión de tipos de documentos.

3.7. Gestión de banco y cuentas bancarias

Su objetivo es registrar y mantener la información de las entidades bancarias y las cuentas bancarias de la empresa para la gestión de flujos de efectivo y conciliaciones, asociando las operaciones financieras con su representación contable mediante funcionalidades que incluyen la creación de bancos, la asociación de cada banco con su respectiva cuenta bancaria y cuenta contable, la modificación de información de bancos y cuentas, y la eliminación de bancos. Este submódulo se conecta con los módulos de Tesorería y Cartera donde los pagos y cobros que involucren movimientos bancarios se registran contra estas cuentas, se integra con los Reportes Financieros - Estados Financieros donde el estado de flujos de efectivo se nutre de la información de las cuentas bancarias y se relaciona con los módulos Contable-Comercial y Contable Cartera donde los asientos contables relacionados con movimientos bancarios se generan con base en esta configuración, constituyendo así el repositorio fundamental que vincula las operaciones bancarias físicas con su registro contable correspondiente en el sistema. A continuación se muestran de manera gráfica las relaciones correspondientes del submódulo de Gestión de bancos y cuentas bancarias.

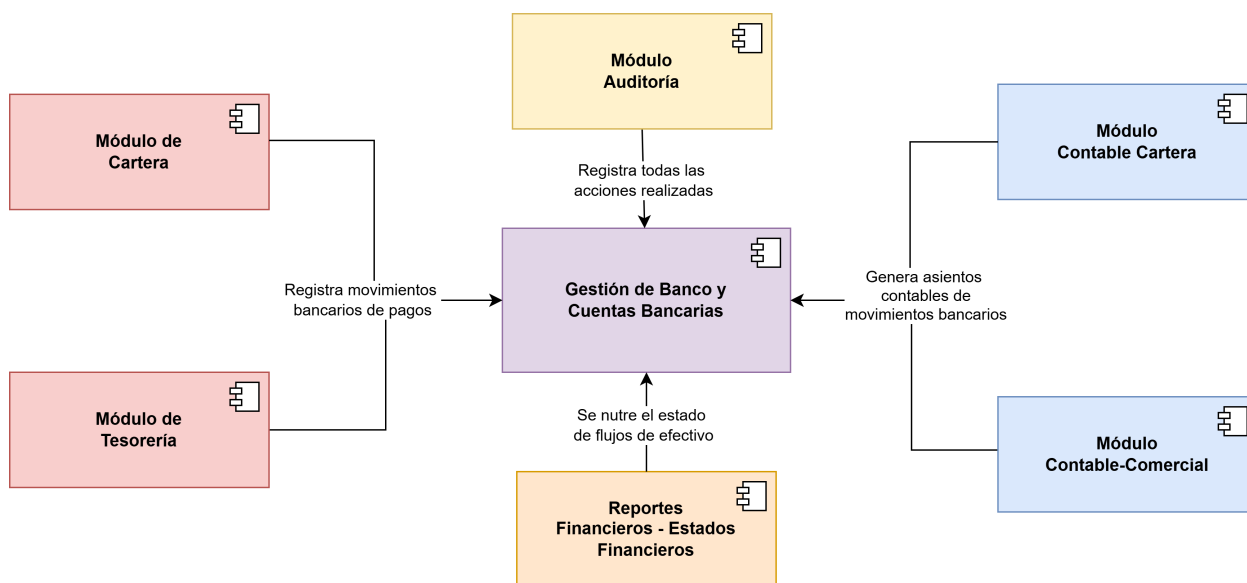


Figura 3-8.: Relación del submódulo gestión de bancos y cuentas bancarias.

3.8. Gestión de centros de costo

Su objetivo es permitir la segmentación de las operaciones contables y presupuestales por áreas funcionales, unidades organizacionales o proyectos para un análisis gerencial de resultados más detallado, proporcionando funcionalidades que comprenden la creación, modificación y eliminación de centros de costo y la habilitación de la selección de un centro de costo al registrar una operación contable o comercial. Este submódulo se conecta con el Módulo Comercial donde las ventas y compras pueden asignarse a centros de costo, se integra con el Módulo de Tesorería permitiendo que los egresos puedan asignarse a centros de costo, se relaciona con los módulos Contable-Comercial y Contable Cartera donde los asientos contables pueden incluir la dimensión de centro de costo, y se vincula con los Reportes Financieros tanto de Libros Auxiliares como de Estados Financieros donde los reportes pueden filtrarse y presentarse por centro de costo, constituyendo así el elemento dimensional que enriquece el análisis gerencial y la trazabilidad de las operaciones por unidades organizacionales específicas. A continuación se muestran de manera gráfica las relaciones correspondientes del submódulo de Gestión de centros de costo.

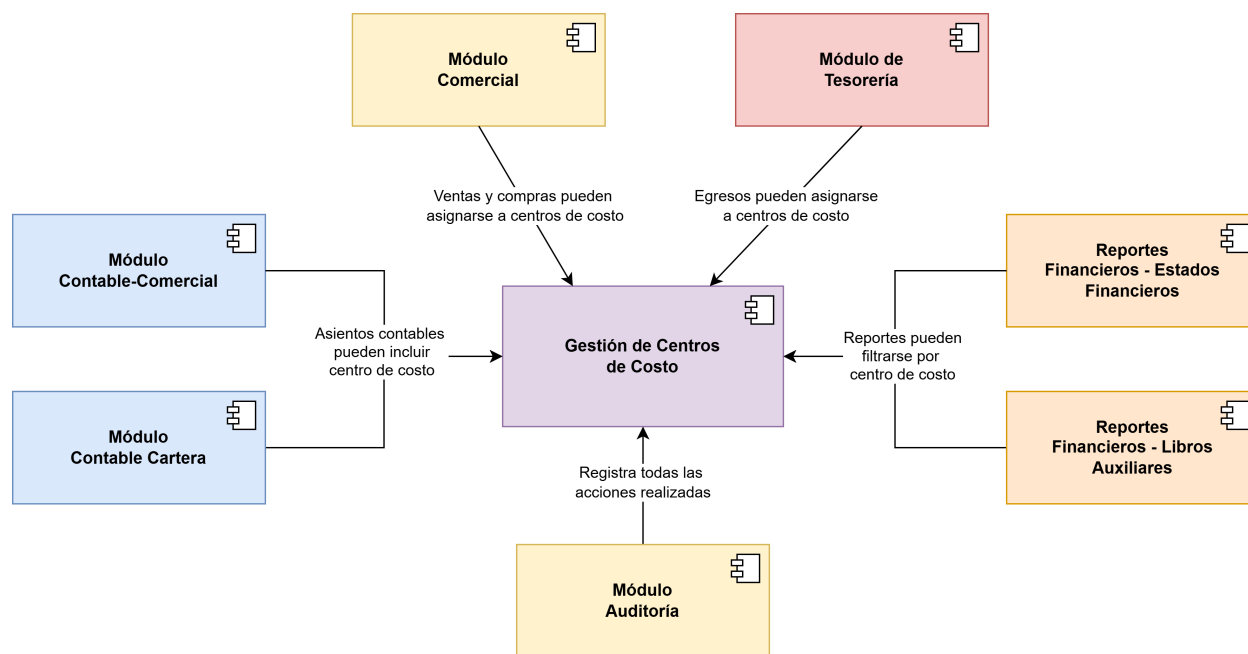


Figura 3-9.: Relación del submódulo gestión de centros de costo.

3.9. Gestión de centro de ayuda

Contribuye a la comprensión y el uso del sistema por parte de los usuarios, permitiendo la gestión de mensajes informativos y explicativos a través de funcionalidades que incluyen la creación, modificación de contenido y eliminación de cuadros de ayuda explicativos. Este módulo establece relaciones transversales con todos los módulos funcionales del sistema, ya que los cuadros de ayuda pueden aparecer en cualquier módulo para proporcionar ayuda contextual, explicaciones de conceptos o advertencias, los cuales se visualizan en un sitio web tipo blog donde se muestran las ayudas creadas, mejorando la experiencia del usuario al ofrecer información relevante y oportuna durante la navegación y operación de las diferentes funcionalidades del sistema, constituyendo así un elemento de soporte que contribuye a la usabilidad y accesibilidad de la plataforma en su conjunto. A continuación se muestran de manera gráfica las relaciones correspondientes del submódulo de Gestión de cuadros de ayuda.

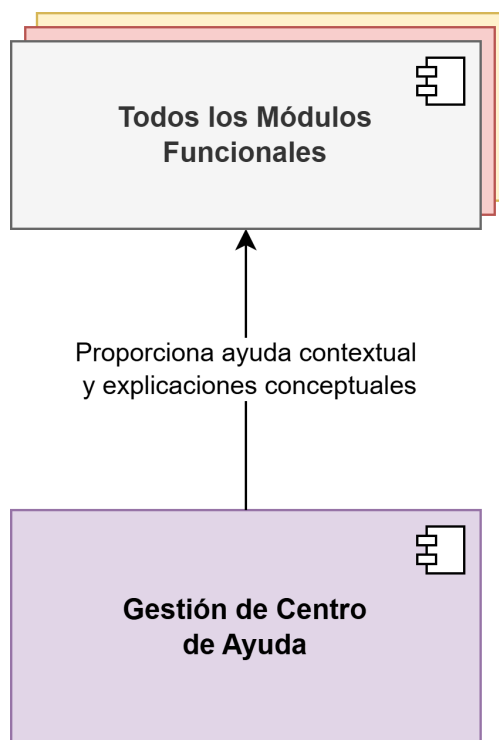


Figura 3-10.: Relación del submódulo gestión de cuadros de ayuda.

3.10. Gestión de calendario contable

Es el componente temporal que estructura y controla el ciclo de vida operativo del sistema. Su objetivo es definir periodos abiertos y cerrados para garantizar que todas las transacciones se registren en el momento correcto, contribuyendo a la integridad, trazabilidad y validez legal de la información financiera. Al permitir que el administrador abra o bloquee meses, evita registros fuera de periodos autorizados, soporta el cierre contable mensual y genera la base auditada para reportes oficializados. Este calendario valida fechas en inventarios, facturación, tesorería y cartera antes de permitir registros; restringe la generación de asientos contables en periodos cerrados y filtra reportes financieros por fechas activas. A continuación se muestran de manera gráfica las relaciones correspondientes del submodulo de Gestión de calendario contable.

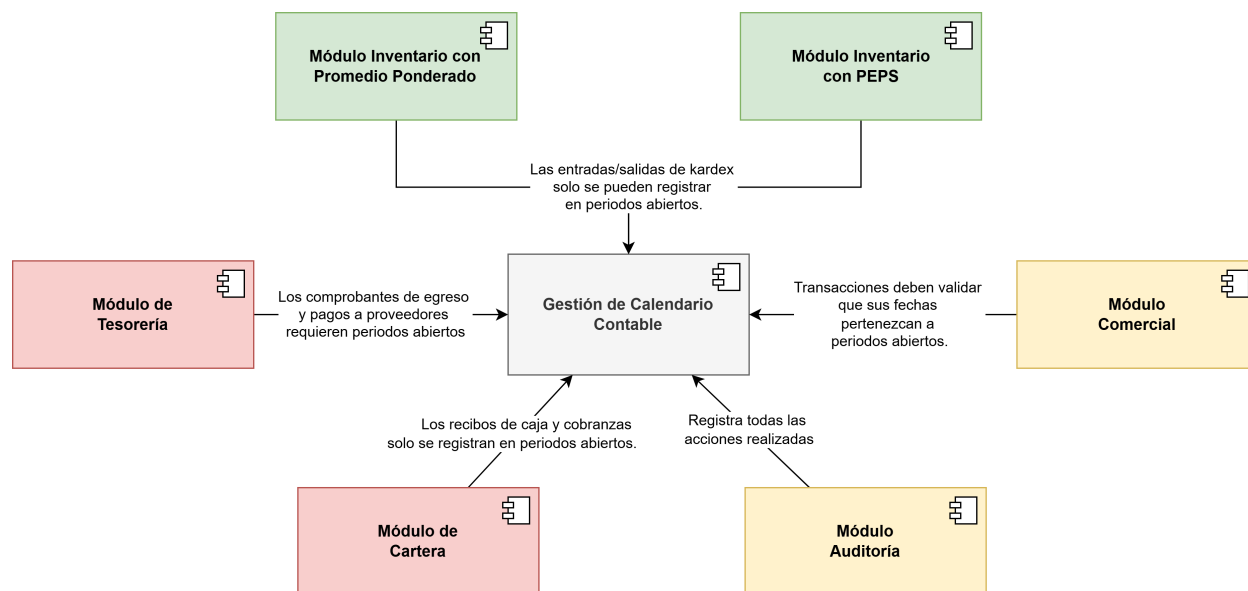


Figura 3-11.: Relación del submódulo gestión de calendario contable.

Capítulo 4. Ingeniería de requisitos

La ingeniería de requisitos es una etapa fundamental en un proyecto de software, ya que permite identificar, analizar y documentar las necesidades de un sistema antes de su implementación. Un levantamiento adecuado de requisitos permite que el resultado cumpla las expectativas del cliente. Es por esto que en este capítulo se describe el proceso llevado a cabo para la especificación de los requerimientos funcionales y no funcionales del módulo de configuración del “Software contable para uso educativo en el programa de Contaduría Pública de la Universidad del Cauca”.

Durante la ingeniería de requisitos llevada a cabo en el proyecto se realizaron diferentes actividades con el fin de obtener una lista de tareas congruente con los requisitos del software contable educativo. Para hacer posible este proceso fue fundamental la comunicación constante con los principales interesados, docentes del programa de Contaduría Pública, el asesor empresarial Mauro Andrés Sánchez Muñoz y el director de la práctica profesional PhD. Wilson Libardo Pantoja Yopez, quienes muy amablemente reservaron espacios en sus agendas para atender las inquietudes resultantes. Esto permitió una retroalimentación constante y sumado a la flexibilidad de la metodología de trabajo híbrida adoptada, fue posible ir ajustando los requisitos en cada encuentro, incluso en etapas posteriores del desarrollo.

También se debe tener en cuenta que el análisis previo descrito en el “Capítulo 3. Descripción de los submódulos a implementar”, en el cual se identificó los submódulos que componen el módulo de configuración, permitió tener un contexto más amplio sobre los procesos y actividades involucradas en la gestión contable educativa. Lo anterior claramente benefició la recolección y especificación de requerimientos, dado que fue posible avanzar con mayor seguridad y velocidad en la definición de funcionalidades para cada submódulo.

Sin embargo, a pesar de la disposición de los interesados, se presentaron dificultades relacionadas con la coordinación de horarios y las múltiples ocupaciones de los participantes. En ocasiones, los docentes del programa de Contaduría Pública no disponían de tiempo suficiente para el proyecto debido a sus compromisos académicos, evaluaciones y actividades administrativas. A pesar de ello, se logró ajustar las actividades y el trabajo requerido para el levantamiento de requisitos, superando los inconvenientes mediante la flexibilidad horaria característica del enfoque freelance adoptado.

4.1. Actividades desarrolladas para la ingeniería de requisitos

Las actividades desarrolladas para llevar a cabo la ingeniería de requisitos para este proyecto son las siguientes.

4.1.1. Reuniones con los interesados

Como previamente se ha mencionado, durante el proyecto fue de suma importancia los encuentros presenciales y virtuales con los interesados, donde se resolvieron dudas sobre aspectos contables y pedagógicos y se obtuvo retroalimentación constante la cual moldeaba el trabajo que se realizaba. También en ocasiones se compartió documentación sobre las actividades llevadas a cabo en el programa de Contaduría Pública, como planes de estudio, casos de uso contables y ejemplos de ejercicios prácticos útiles para el proceso. Se debe notar que, dada la naturaleza técnica del proyecto, la comunicación presencial y por videoconferencia resultó más efectiva que los medios digitales asíncronos para la discusión de conceptos contables complejos.

4.1.2. Construcción de prototipos

La construcción de prototipos fue fundamental para no dirigir el diálogo con los docentes de forma exclusivamente técnica, sino establecer una conversación apoyada visualmente por los prototipos de interfaz. Los interesados aportaron sus comentarios sobre cambios en las pantallas de gestión de cuentas contables, terceros, productos, impuestos y demás submódulos identificados anteriormente, evaluando su adecuación para el entorno educativo. Asimismo se realizaron negociaciones sobre la complejidad de las funcionalidades según el nivel académico de los estudiantes. Los cambios producto de estas negociaciones eran presentados como apertura en la siguiente reunión para poder continuar el proceso iterativo.

4.1.3. Definición de historias de usuario

La conversación con los docentes se llevó a cabo de forma pedagógica y orientada a casos de uso contables reales, como se mencionó anteriormente. Sin embargo, para el proceso de desarrollo del proyecto es fundamental tener documentación estructurada que defina claramente las funcionalidades que requiere el módulo de configuración del software contable educativo. Es por esto que las historias de usuario como artefacto para el levantamiento de requerimientos fueron realizadas de forma paralela a la construcción de prototipos, considerando

tanto los aspectos técnicos del desarrollo como los objetivos pedagógicos del programa de Contaduría Pública, estas historias fueron ajustadas según la retroalimentación brindada por el cliente en cada encuentro, asegurando que cada submódulo cumpliera con los requisitos educativos específicos.

4.1.4. Validación de prototipos

Como ya fue mencionado, la validación de prototipos se realizó en la apertura de cada encuentro con el cliente, de modo que se tuviera seguridad de que las interfaces diseñadas fueran intuitivas para estudiantes de contabilidad y cumplieran con los lineamientos de usabilidad establecidos por la División TIC de la Universidad del Cauca. Se evaluó especialmente la claridad de los formularios de gestión de cada submódulo y la coherencia con los procesos contables estándar.

4.1.5. Validación de historias de usuario

La validación de las historias de usuario se realizó en conjunto con la validación de prototipos, estas fueron ajustadas en cada encuentro, considerando tanto los aspectos técnicos como los requisitos pedagógicos específicos del programa de Contaduría Pública. En este proceso, se prestó especial atención a que cada funcionalidad contribuyera efectivamente al aprendizaje de conceptos contables fundamentales y permitiera la simulación de escenarios empresariales reales.

4.2. Identificación de stakeholders

Para identificar los stakeholders del proyecto fue fundamental la colaboración con el asesor empresarial y el director de la práctica profesional. Es por esto que se logra la identificación de los interesados como a continuación se describe:

Tabla 4-1.: Stakeholders del proyecto

Interesado	Cargo
Mauro Andrés Sánchez Muñoz	Asesor de la empresa
Viviana Patricia Narváez Castillo	Docente del programa de Contaduría Pública
Juan Ignacio Oviedo Pino	Docente del programa de Contaduría Pública
Desarrolladores de otros módulos del sistema	Colaboradores técnicos para integración
Estudiantes del programa de Contaduría Pública	Usuarios finales del sistema
División TIC Universidad del Cauca	Supervisores técnicos y de lineamientos

4.3. Prototipos

La construcción de prototipos fue esencial para una comunicación clara con el asesor empresarial, ya que al permitir visualizar un acercamiento de lo que serían las interfaces del módulo de configuración se obtenía retroalimentación constante y rápida sobre aspectos pedagógicos y de usabilidad. Esto ha permitido realizar un refinamiento de la especificación de requerimientos en el formato de historias de usuario, este último también ha contribuido en sentido contrario a detallar más cuidadosamente los prototipos, generando así un círculo virtuoso de mejora continua.

Los prototipos se enfocaron en representar las interfaces de los submódulos que componen el módulo de configuración, cada prototipo consideró los lineamientos de diseño UX/UI establecidos por la División TIC de la Universidad del Cauca, incluyendo la paleta de colores institucional y las fuentes tipográficas especificadas.

Para el diseño de los prototipos se hizo necesario el uso de una herramienta flexible y de fácil manejo, en la que se pudiera plasmar rápidamente el concepto de las interfaces del módulo de configuración, la herramienta elegida fue Figma por su comodidad, flexibilidad y fácil manejo para el diseño de interfaces web, además de permitir la colaboración en tiempo real con los interesados del proyecto. Por tanto, si se desea explorar en su totalidad los prototipos desarrollados, es preciso dirigirse al proyecto Figma en el siguiente [enlace](#), donde se encuentran compilados todos los diseños de las interfaces del módulo de configuración.

Igualmente, se incluyen a continuación ejemplos seleccionados de los prototipos desarrollados, evitando así saturar el documento y optimizando la comprensión visual.

4.3.1. Vista de catálogo de cuentas

La figura 4-1 muestra el prototipo asociado al catálogo de cuentas, que establece la estructura maestra de clasificación contable donde se definen, organizan y gestionan todas las cuentas que registrarán las transacciones financieras del sistema. Configura el esquema contable maestro del sistema a través de una interfaz que permite gestionar jerárquicamente la estructura de clasificación financiera. Desde este módulo central, el administrador puede crear, importar o exportar cuentas codificadas numéricamente del 1 al 9, organizadas según naturaleza contable (Activo, Pasivos, Patrimonio, Ingresos, Gastos, Costos de transformación y Cuentas de orden), con control de estado activo/inactivo para mantener vigente el plan de cuentas.

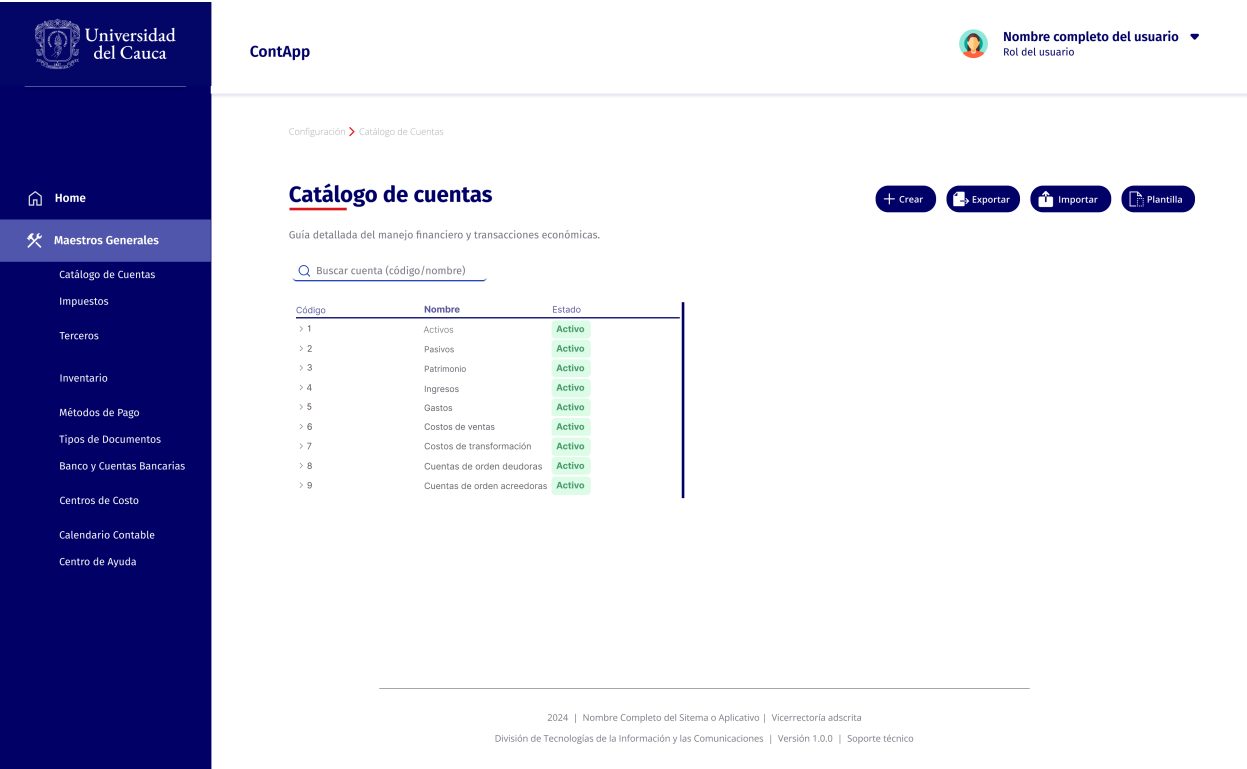


Figura 4-1.: Prototipo vista catálogo de cuentas.

4.3.2. Vista de inventario de productos

La figura 4-2 muestra el prototipo asociado al inventario de productos establece el catálogo maestro de artículos donde se centraliza la gestión de insumos, productos terminados y mercancías a través de una interfaz que permite crear, importar, exportar y administrar registros con atributos clave como referencia, nombre, descripción, costo, cantidad, unidad de medida, categoría, tipo de producto y estado activo.

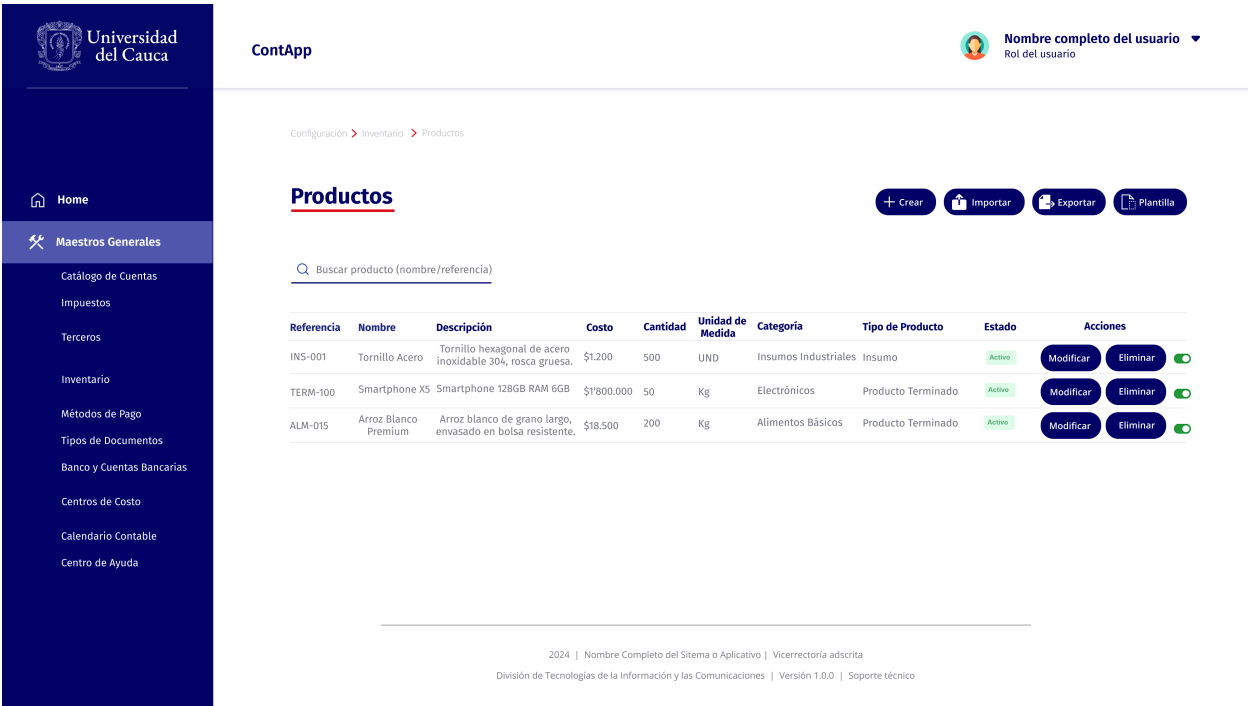


Figura 4-2.: Prototipo vista inventario de productos.

4.3.3. Vista de terceros

La figura 4-3 muestra el prototipo asociado a terceros, configura el registro maestro de entidades externas a través de una interfaz que permite crear, importar, exportar y administrar personas naturales y jurídicas, capturando atributos esenciales como identificación, tipo de entidad, correo electrónico y estado activo. Incorpora la funcionalidad específica de procesar archivos PDF del RUT para automatizar la creación de registros, validando duplicados y formatos de datos requeridos. Este módulo opera como repositorio centralizado que estructura y mantiene actualizada la base de datos de clientes, proveedores y otros actores, habilitando búsquedas dinámicas por nombre o numero de identificación.

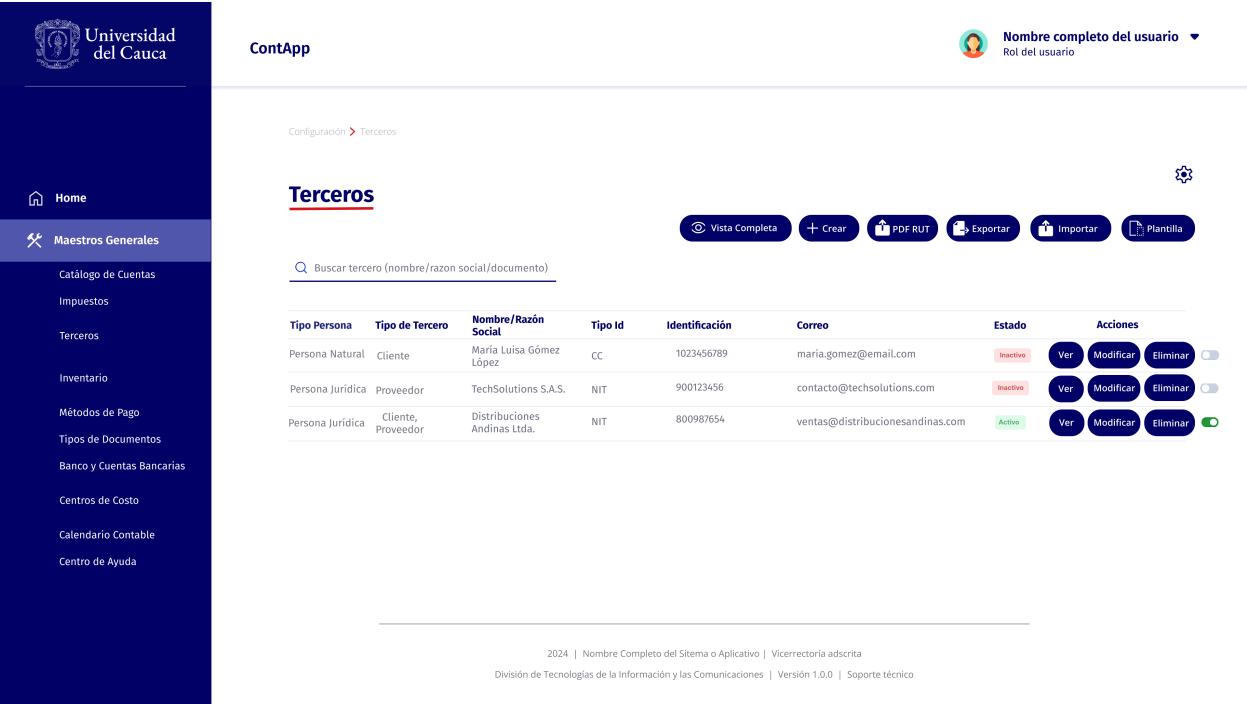


Figura 4-3.: Prototipo vista de terceros.

4.3.4. Vista de impuestos

La figura 4-4 muestra el prototipo asociado a impuestos encargado de la configuracion maestra de tarifas mediante una interfaz que permite crear, modificar y eliminar registros, asignando descripciones, porcentajes y las cuentas contables específicas para compras y ventas. Centraliza la parametrización de tributos como IVA y retenciones, validando la existencia de las cuentas asociadas.

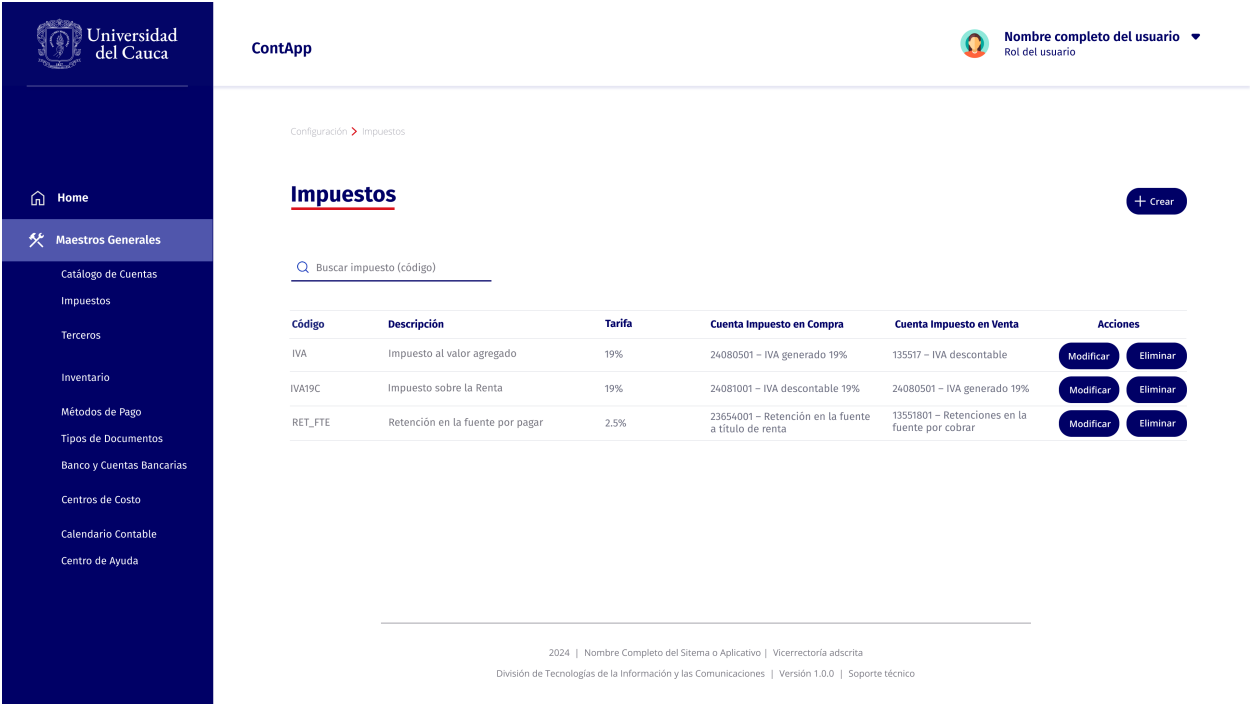


Figura 4-4.: Prototipo vista de impuestos.

4.4. Requisitos funcionales

La lista de tareas del proyecto refleja las necesidades identificadas y se estructura mediante historias épicas, las cuales a su vez se desglosan en historias de usuario. Las historias épicas agrupan funcionalidades o tareas complejas que no pueden abordarse en una sola iteración, mientras que las historias de usuario detallan estas épicas en descripciones manejables que permiten su desarrollo independiente en una iteración. A continuación, se presenta la tabla con la lista de tareas del proyecto.

Tabla 4-2.: Lista de tareas del proyecto: historias épicas y de usuario

Épica		Historia de Usuario Asociada	
ID Épica	Nombre Épica	ID Historia	Nombre Historia
HE-01	Gestionar el catálogo de cuentas	HE-01-HU01	Crear cuentas contables
		HE-01-HU02	Modificar cuentas contables
		HE-01-HU03	Eliminar cuentas contables
		HE-01-HU04	Importar catálogo de cuentas desde archivo plano
		HE-01-HU05	Exportar catálogo de cuentas en un archivo plano
		HE-01-HU06	Consultar cuentas contables
		HE-01-HU07	Activar/Desactivar cuentas contables
HE-02	Gestionar las tarifas de los impuestos	HE-02-HU01	Crear nuevas tarifas de impuestos
		HE-02-HU02	Modificar tarifas de impuestos
		HE-02-HU03	Eliminar tarifas de impuestos
		HE-02-HU04	Consultar tarifas de impuestos
		HE-02-HU05	Activar/Desactivar tarifas de impuestos
HE-03	Gestionar la información de terceros	HE-03-HU01	Crear terceros
		HE-03-HU02	Crear terceros desde PDF (RUT)
		HE-03-HU03	Modificación de terceros
		HE-03-HU04	Activar/Desactivar terceros
		HE-03-HU05	Importar terceros desde archivo plano
		HE-03-HU06	Exportar terceros en un archivo plano
		HE-03-HU07	Consultar terceros
HE-04	Gestionar el catálogo de productos del inventario	HE-04-HU01	Crear productos en el inventario
		HE-04-HU02	Modificar productos en el inventario
		HE-04-HU03	Eliminar productos en el inventario
		HE-04-HU04	Importar productos desde archivo plano
		HE-04-HU05	Exportar productos en un archivo plano
		HE-04-HU06	Consultar productos en el inventario

Tabla 4-2 Continuación: Lista de tareas del proyecto

Épica		Historia de Usuario Asociada	
HE-05	Gestionar el centro de ayuda	HE-05-HU01	Crear centros de ayuda explicativos
		HE-05-HU02	Modificar contenido en centros de ayuda
		HE-05-HU03	Eliminar centros de ayuda
		HE-05-HU04	Consultar centros de ayuda
		HE-05-HU05	Activar/Desactivar centros de ayuda
HE-06	Gestionar los métodos de pago	HE-06-HU01	Crear métodos de pago
		HE-06-HU02	Modificar métodos de pago
		HE-06-HU03	Eliminar métodos de pago
		HE-06-HU04	Consultar métodos de pago
		HE-06-HU05	Activar/Desactivar métodos de pago
HE-07	Gestionar los tipos de documento	HE-07-HU01	Crear tipos de documento
		HE-07-HU02	Modificar tipos de documento
		HE-07-HU03	Eliminar tipos de documento
		HE-07-HU04	Consultar tipos de documentos
		HE-07-HU05	Activar/Desactivar tipos de documentos
HE-08	Gestionar los bancos y sus cuentas bancarias	HE-08-HU01	Crear bancos
		HE-08-HU02	Modificar bancos
		HE-08-HU03	Eliminar bancos
		HE-08-HU04	Consultar bancos
		HE-08-HU05	Activar/Desactivar bancos
HE-09	Gestionar los centros de costos	HE-09-HU01	Crear centros de costo
		HE-09-HU02	Modificar centros de costo
		HE-09-HU03	Eliminar centros de costo
		HE-09-HU04	Consultar centros de costo
		HE-09-HU05	Activar/Desactivar centros de costo
HE-10	Gestionar el calendario contable interactivo	HE-10-HU01	Abrir/cerrar año contable
		HE-10-HU02	Abrir/cerrar mes contable
		HE-10-HU03	Abrir/cerrar fecha específica
		HE-10-HU04	Consultar periodos contables

Capítulo 5. Arquitectura del sistema

Capítulo 6. Patrones de diseño

Capítulo 7. Pruebas realizadas a la aplicación web

Capítulo 8. Conclusiones

A. Anexo: Nombrar el anexo A de acuerdo con su contenido

Los Anexos son documentos o elementos que complementan el cuerpo de la tesis o trabajo de investigación y que se relacionan, directa o indirectamente, con la investigación, tales como acetatos, cd, normas, etc.

B. Anexo: Nombrar el anexo B de acuerdo con su contenido

A final del documento es opcional incluir índices o glosarios. Éstos son listas detalladas y especializadas de los términos, nombres, autores, temas, etc., que aparecen en el mismo. Sirven para facilitar su localización en el texto. Los índices pueden ser alfabéticos, cronológicos, numéricos, analíticos, entre otros. Luego de cada palabra, término, etc., se pone coma y el número de la página donde aparece esta información.

C. Anexo: Nombrar el anexo C de acuerdo con su contenido

MANEJO DE LA BIBLIOGRAFÍA: la bibliografía es la relación de las fuentes documentales consultadas por el investigador para sustentar sus trabajos. Su inclusión es obligatoria en todo trabajo de investigación. Cada referencia bibliográfica se inicia contra el margen izquierdo.

La NTC 5613 establece los requisitos para la presentación de referencias bibliográficas citas y notas de pie de página. Sin embargo, se tiene la libertad de usar cualquier norma bibliográfica de acuerdo con lo acostumbrado por cada disciplina del conocimiento. En esta medida es necesario que la norma seleccionada se aplique con rigurosidad.

Es necesario tener en cuenta que la norma ISO 690:1987 (en España, UNE 50-104-94) es el marco internacional que da las pautas mínimas para las citas bibliográficas de documentos impresos y publicados. A continuación se lista algunas instituciones que brindan parámetros para el manejo de las referencias bibliográficas:

Bibliografía

- [1] Y. Blount et al. “Integrating enterprise resource planning (SAP) in the accounting curriculum: a systematic literature review and case study”. En: *Accounting Education* 25.2 (2016), págs. 185-202. DOI: 10.1080/09639284.2016.1138136.
- [2] E. Boulianne. “Impact of accounting software utilization on students’s knowledge acquisition: An important change in accounting education”. En: *Journal of Accounting and Organizational Change* 10.1 (2014), págs. 22-48. DOI: 10.1108/JAOC-12-2011-0064.
- [3] Universidad del Cauca. *Contaduría Pública*. URL: <https://portal.unicauca.edu.co/versionP/oferta-academica/programas-de-pregrado/contaduria-publica>.
- [4] Equipo de Cloud Gestion. *Configurar el módulo de contabilidad en un Software ERP — cloudgestion.com*. <https://www.cloudgestion.com/blog/software/configurar-modulo-contabilidad/>. [Accessed 05-05-2025].
- [5] TechTarget Contributor. *What Is Hibernate? Definition from TheServerSide*. URL: <https://www.theserverside.com/definition/Hibernate>.
- [6] L. Daff. “Employers’s perspectives of accounting graduates and their world of work: software use and ICT competencies”. En: *Accounting Education* 30.5 (2021), págs. 495-524. DOI: 10.1080/09639284.2021.1935282.
- [7] *Desarrollo iterativo e incremental*. <https://proyectosagiles.org/desarrollo-iterativo-incremental/>. [Accessed 14-07-2025].
- [8] División de Tecnologías de la Información y las Comunicaciones. *Lineamientos de diseño UX/UI para proyectos de desarrollo de software en la División TIC de la Universidad del Cauca*. es. 2024.
- [9] División de Tecnologías de la Información y las Comunicaciones. *Lineamientos normativos y pautas de usabilidad para el diseño de interfaces en los proyectos de desarrollo software en la División TIC de la Universidad del Cauca*. es. Jun. de 2024.
- [10] División TIC, Universidad del Cauca. *Lineamientos de Desarrollo, Pasantías Y Prácticas Profesionales*. es. 2024.
- [11] Manager ERP. *¿Cómo configurar un ERP? - Manager Software — manager.cl*. <https://www.manager.cl/blog/como-configurar-un-erp/>. [Accessed 05-05-2025].

- [12] *ERP Implementation: Steps, Challenges & Best Practices - Apty* — *apty.ai*. <https://apty.ai/blog/erp-implementation/>. [Accessed 05-05-2025].
- [13] *Framework - Wikipedia, la enciclopedia libre* — *es.wikipedia.org*. <https://es.wikipedia.org/wiki/Framework>. [Accessed 15-07-2025].
- [14] Software de Gestion myGESTION. *¿Qué es la parametrización de un ERP?* — *mygestion.com*. <https://www.mygestion.com/blog/parametrizacion-erp>. [Accessed 05-05-2025].
- [15] S. Gould. *Disrupting the Accountancy Profession* | IFAC. 2017. URL: <https://www.ifac.org/knowledge-gateway/developing-accountancy-profession/discussion/disrupting-accountancy-profession> (visitado 06-12-2024).
- [16] grokkeepcoding. *Eureka Server: su papel en la arquitectura de microservicios*. <https://keepcoding.io/blog/que-es-eureka-server-en-la-arq-de-microservicio/>. [Accessed 12-07-2025].
- [17] *Home* — *c4model.com*. <https://c4model.com/>. [Accessed 18-07-2025].
- [18] Emprendedores de Hoy. *¿CEESA, S.A. impulsa la eficiencia empresarial con su software ERP para pymes?* — *diariosigloxxi.com*. <https://www.diariosigloxxi.com/texto-diario/mostrar/5275054/ceesa-sa-impulsa-eficiencia-empresarial-software-erp-pymes>. [Accessed 05-05-2025].
- [19] A. Huczynski y S. P. Johnston. “Engineering students’ use of computer assisted learning (CAL)”. En: *European Journal of Engineering Education* 30.2 (2005), págs. 287-298. DOI: 10.1080/03043790500087530.
- [20] R. L. Humphrey y D. F. Beard. “Faculty perceptions of online homework software in accounting education”. En: *Journal of Accounting Education* 32.3 (2014), págs. 238-258. DOI: 10.1016/j.jaccedu.2014.06.001.
- [21] IBM. *¿Qué son las pruebas de software?* | IBM — *ibm.com*. <https://www.ibm.com/es-es/think/topics/software-testing>. [Accessed 15-07-2025]. 2021.
- [22] *IBM Business Automation Workflow 23.x*. Feb. de 2024. URL: <https://www.ibm.com/docs/en/baw/23.x?topic=formats-javascript-object-notation-json-format>.
- [23] *Introduction to Tailwind CSS - GeeksforGeeks*. Oct. de 2024. URL: <https://www.geeksforgeeks.org/introduction-to-tailwind-css/>.
- [24] S. Kanapathippillai, A. Shamlee Hasheem y S. Dellaportas. “The impact of a computerised consolidation accounting package (CCAP) on student performance”. En: *Asian Review of Accounting* 20.1 (2012), págs. 4-19. DOI: 10.1108/13217341211224691.
- [25] Jose Manuel Rosa Moncayo. *¿Qué es REST: Conoce su potencia* | *OpenWebinars*. Mayo de 2018. URL: <https://openwebinars.net/blog/que-es-rest-conoce-su-potencia/>.

- [26] Yanina Muradas. *QuÃ© es Spring Framework y por quÃ© usarlo* / *OpenWebinars*. Jun. de 2018. URL: <https://openwebinars.net/blog/conoce-que-es-spring-framework-y-por-que-usarlo/>.
- [27] A. Novak, K. Åžager e I. BariÅŸiÅŸ. "Innovation in accounting education: The impact of information technology on teaching methods". En: *Proceedings of the European Conference on Innovation and Entrepreneurship, ECIE*. 2021, págs. 660-667. DOI: 10.34190/EIE.21.142.
- [28] M. Osmani, N. Hindi y V. Weerakkody. "Incorporating information communication technology skills in accounting education". En: *International Journal of Information and Communication Technology Education* 16.4 (2020), págs. 100-110. DOI: 10.4018/IJICTE.2020100107.
- [29] *QuÃ© es un freelance* / *GuÃa del trabajador independiente* — *mividalaboral.com*. <https://mividalaboral.com/freelance/>. [Accessed 06-05-2025].
- [30] Suryaansh Rathinam. "Analysis and Comparison of Different Frontend Frameworks". En: *Applications and Techniques in Information Security*. Ed. por Srikanth Prabhu, Shiva Raj Pokhrel y Gang Li. Singapore: Springer Nature Singapore, 2023, págs. 243-257.
- [31] A. Sangster. "Computer-based instruction in accounting education". En: *Accounting Education* 1.1 (1992), págs. 13-32. DOI: 10.1080/09639289200000003.
- [32] B. Sloan. "In search of excellence in learning: The strategic value of computer-assisted learning (CAL)". En: *Construction Management and Economics* 13.5 (1995), págs. 435-439. DOI: 10.1080/01446199500000049.
- [33] Kanban University. *La Guia Oficial del Metodo Kanban*. https://resources.kanban.university/wp-content/uploads/2021/08/The-Official-Kanban-Guide-Spanish_A4. [Accessed 05-05-2025].
- [34] R. M. A. Wadi y R. J. Jaber. "The Use of Information Technology in Accounting Education, Evidence from Jordan". En: *Lecture Notes in Networks and Systems: Vol. 194 LNNS*. 2021, págs. 801-811. DOI: 10.1007/978-3-030-69221-6_61.
- [35] M. Wali y L. Ahmad. "Computer Assisted Learning (CAL): A Learning Support System Solution". En: *Webology* 18.1 (2021), págs. 299-314. DOI: 10.14704/WEB/V18I1/WEB18090.
- [36] W. M. N. Wan Mohd Nori et al. "The use of accounting software for effective teaching and learning introductory accounting: A pedagogical discussion". En: *Social Sciences (Pakistan)* 11.13 (2016), págs. 3315-3320. DOI: 10.3923/sscience.2016.3315.3320.
- [37] V. F. Willis. "A model for teaching technology: Using Excel in an accounting information systems course". En: *Journal of Accounting Education* 36 (2016), págs. 87-99. DOI: 10.1016/j.jaccedu.2016.05.002.

-
- [38] ¿Qué es Java? - Explicación del lenguaje de programación Java - AWS. URL: <https://aws.amazon.com/es/what-is/java/>.
- [39] ¿Qué es JPA y ORM? URL: <https://platzi.com/clases/8048-java-sql/64356-que-es-jpa-y-orm/>.