



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 3

INFORME TÉCNICO

Título: Sistema de gestión para el almacén de libros de la Facultad 1.

Autora: Lecsi Hernández Mayedo.

Tutora: Msc. Geidis Sánchez Michel.

La Habana, noviembre 2020

“Año 62 de la Revolución”

RESUMEN

El Vicedecanato de Economía y Administración de la Facultad 1 realiza el control y gestión del almacén de libros de forma manual lo que conduce en ocasiones a múltiples problemas en cuanto a tiempo y veracidad de la información, por errores humanos, por lo que existe la necesidad de desarrollar una solución que integre e informatice estos procesos usando las modernas tecnologías computacionales. La presente investigación se enmarca en la comprensión, análisis e interpretación de las características de la gestión de los procesos de entrega y devolución de libros en el Vicedecanato de Economía y Administración de la Facultad 1. El objetivo de esta investigación es desarrollar un sistema que garantice gestionar la entrega y devolución de libros en el Vicedecanato de Economía y Administración de la Facultad 1.

En el desarrollo de la investigación se realiza un estudio del estado del arte de los temas relacionados con la gestión de los procesos de entrega y devolución de libros y la gestión del almacén de libros en el Vicedecanato de Economía y Administración de la Facultad 1. Se utiliza RUP como metodología de desarrollo para guiar la investigación. Se realiza una descripción de los requisitos funcionales y no funcionales que expresan las necesidades del cliente. Se hace el análisis y modelado del sistema en función de la metodología seleccionada.

Palabras clave: almacén, control, gestión, libros, sistema.

ABSTRACT

The Vice Dean of Economics and Administration of Faculty 1 controls and manages the book store manually, which sometimes leads to multiple problems in terms of time and accuracy of the information, due to human errors, so there is a need to develop a solution that integrates and computerizes these processes using modern computational technologies. This research is part of the understanding, analysis and interpretation of the characteristics of the management of the delivery and return of books in the Vice-Dean of Administration and Economics of the Faculty 1. The objective of this research is to develop a system that guarantees manage the delivery and return of books in the Vice-Dean of Economics and Administration of the Faculty 1.

In the development of the research, conducts a study of the art on topics related to the management of the processes of delivery and return of books and the management of the book store is carried out in the Vice-Dean of Economics and Administration of the Faculty 1. RUP is used as the development methodology to guide the research. A description of the functional and non-functional requirements that express the customer's needs is made. The analysis and modeling of the system is carried out according to the selected methodology.

Keywords: warehouse, control, management, books, system.

ÍNDICE

INTRODUCCIÓN.....	5
1 Desarrollo	9
1.1 Conceptos asociados	9
1.2 Estudio de Sistemas Homólogos.....	13
1.2.1 Estudio de sistemas homólogos en el ámbito internacional	13
1.2.2 Estudio de sistemas en el ámbito nacional	14
1.3 Ambiente de desarrollo.....	17
1.3.1 Metodología de desarrollo	17
1.3.2 Herramientas	17
2 Propuesta de solución	20
2.1.1 Diagrama de casos de uso del negocio	20
2.1.2 Diagrama de actividades	21
2.2 Especificación de requisitos	23
2.3 Requisitos no funcionales.....	28
2.3.1 Diagrama de caso de uso del sistema	29
2.3.2 Diagrama de clases del análisis.....	39
2.3.3 Diagrama de clases del diseño	41
2.4 Arquitectura de la solución	44
2.5 Patrones de diseño	46
2.5.1 Diagrama de colaboración	46
2.6 Modelo Entidad- Relación	48
2.7 Estructura del sistema	48
2.8 Estándar de codificación de Symfony.....	49
2.9 Diagrama de despliegue.....	50
2.10 Pruebas de software	51
2.11 Verificación y validación de funcionalidades	51
2.12 Estrategia de pruebas	52
2.12.1 Niveles de prueba	52

CONCLUSIONES.....	53
Bibliografía.....	54

INTRODUCCIÓN

Los libros más que una colección de hojas con letras impresas son evidencia de la evolución del hombre y los relatos que conllevaron a la humanidad al gran avance del que se disfruta hoy en día, por consiguiente, grandes almacenes y bibliotecas surgieron por la necesidad de guardarlos y preservarlos con el objetivo de brindar fácil acceso a registros bibliográficos para posteriores consultas.

El papiro fue en la antigüedad la superficie de escritura más popular de todo el mar Mediterráneo, usada por griegos y romanos, pero originaria de los egipcios los cuales poseían un monopolio virtual sobre el comercio del papiro. Los rollos eran albergados en grandes bibliotecas que actuaban como depósitos de conocimiento y muestra de poder político y económico. La Biblioteca Real de Alejandría contaba con alrededor de medio millón de rollos en su colección, lo que representaba entre el 30 y el 70 por ciento de todos los libros existentes en ese momento. Con el transcurso del tiempo y el evolucionar de la humanidad a medida que crece de forma continua el caudal de información para salvaguardar, crece también la cantidad de almacenes y se edifican bibliotecas más grandes con el fin de darle lugar a cada libro (Una mirada al mundo de las bibliotecas, 2012).

Actualmente es fácil obtener un préstamo de un material bibliográfico ya sea en formato digital o formato duro debido a la gran disponibilidad y asequibilidad con la que se cuenta, dada la distribución de libros que se ha gestionado en centros e instituciones tanto de ocio como educativos.

Cuba a pesar de ser un país que ha sufrido años de bloqueo económico infringido por el gobierno de Estados Unidos ha sido capaz de ofrecer a su población una educación gratuita forjando a la vez valores de conciencia y actuar responsable en cuanto al cuidado de libros y la propiedad social que la Revolución ha puesto a disposición de la sociedad cubana.

Las universidades cubanas cuentan con el material bibliográfico necesario para apoyar a los estudiantes en su formación como futuros profesionales, un ejemplo vigente es la Universidad de las Ciencias Informáticas (UCI), la cual actualmente ofrece titulado para tres carreras universitarias: Ingeniería en Ciencias Informáticas, Ingeniería en Bioinformática y Administración de Redes y Seguridad Informática. Cada nuevo curso escolar la institución abre sus puertas a miles de estudiantes provenientes de todas las provincias del país, e incluso del exterior, además de poseer un amplio claustro de profesores. En la UCI tienen acceso y derecho gratuito a la base material de estudio tanto trabajadores como profesores y estudiantes quienes tienen también el deber de velar por su cuidado durante su uso. Aunque esta casa de altos estudios cuenta con una biblioteca que abarca una amplia gama de temáticas, donde los libros son guardados por catálogos dependiendo de las diferentes categorías existentes, en la Facultad 1, como en el resto de las facultades presentes en la universidad, existe un almacén de libros que registra la

bibliografía básica que responde a la demanda de cada asignatura que se imparte, donde el material bibliográfico se encuentra agrupado por títulos solamente.

El Vicedecanato de Economía y Administración de la Facultad 1 dirige el proceso de préstamo de material bibliográfico, el cual anteriormente se gestionaba directamente en el almacén de la universidad, sin embargo, a día de hoy es un proceso que se dirige desde las facultades. Como es común al inicio de un nuevo curso académico el Vicedecanato, lleva a cabo, cada semestre el proceso de entrega y devolución de libros tanto para estudiantes como profesores, teniendo en cuenta que este es un proceso que se realiza de forma paralela con el curso escolar, ya que es posible solicitar un préstamo, devolver libros o hacer un reporte por pérdida de libros en cualquier momento durante la carrera. Para realizar una solicitud de préstamo de libros el estudiante o trabajador debe proporcionar datos como son: nombre y apellidos, No. Solapín, año en que se realiza el préstamo, grupo al que pertenece en el caso de los estudiantes y área de trabajo a la que pertenece en el caso de los trabajadores. La Facultad 1 cuenta con un local de libros en uso por lo que resulta necesario para el Vicedecanato de Economía y Administración mantener en constancia un inventario donde se relacione la información referente a la cantidad de libros y títulos disponibles, así como la cantidad de libros en préstamo. Para mantener un control sobre el proceso de entrega y devolución de libros, se genera y se mantiene actualizado de forma manual un registro en formato físico por cada estudiante o trabajador al que se le realice el préstamo, el cual lleva por nombre Modelo de Entrega y Devolución de Libros.

El Vicedecanato de Economía y Administración de la Facultad 1 trabaja con 509 estudiantes del Curso Regular Diurno, 156 estudiantes del Curso por Trabajadores y 65 profesores para el primer semestre. Debido a la basta cantidad de estudiantes y trabajadores que atiende el Vicedecanato, se generan altos cúmulos de información para archivar lo cual provoca una lenta gestión del proceso, ya que si un estudiante o profesor solicita un préstamo o devolución de libros, en el caso del estudiante se hace una búsqueda por año académico y luego por grupo hasta localizar el estudiante y en el caso del profesor se hace una búsqueda por área de trabajo a la que está designado. Para tener constancia de la existencia en almacén es necesario contar cada uno de los ejemplares, información que se maneja mediante el uso de Excel, cada vez que se repita el conteo. Debido a que el manejo de la información que se genera se realiza a través del uso de Excel, Microsoft Word y formato duro se dificulta mantener un control sobre la pérdida de datos y extravío de documentos. Por cada documento extraviado es necesario originar un documento nuevo lo cual la mayoría de las veces da lugar a duplicidad de la información.

Para darle solución a la **situación problemática** descrita anteriormente se plantea como **problema** la siguiente interrogante: ¿Cómo gestionar la entrega y devolución de libros en el Vicedecanato de

Economía y Administración de la Facultad 1 para que elimine la duplicidad de información? El **objeto de estudio** tomado es el proceso de entrega y devolución de libros.

El **campo de acción** que abarca la investigación es la gestión de los procesos de entrega y devolución de libros en el Vicedecanato de Administración y Economía de la Facultad 1.

El **objetivo general** que define esta investigación es: Desarrollar un sistema web que gestione la entrega y devolución de libros en el Vicedecanato de Economía y Administración de la Facultad 1 que elimine la duplicidad de información.

Los **objetivos específicos** de este trabajo son los siguientes:

- Definir el marco teórico de la investigación.
- Definir las herramientas y tecnologías a utilizar para el desarrollo de un sistema web que gestione la entrega y devolución de libros en el Vicedecanato de Economía y Administración de la Facultad 1 que elimine la duplicidad de información.
- Analizar los procesos de entrega y devolución de libros en el Vicedecanato de Economía y Administración de la Facultad 1.
- Diseñar un sistema web que gestione la entrega y devolución de libros en el Vicedecanato de Economía y Administración de la Facultad 1 que elimine la duplicidad de información.
- Implementar un sistema web que gestione la entrega y devolución de libros en el Vicedecanato de Economía y Administración de la Facultad 1 que elimine la duplicidad de información.
- Probar que el sistema web diseñado gestiona la entrega y devolución de libros en el Vicedecanato de Economía y Administración de la Facultad 1 que elimine la duplicidad de información.

Para guiar la investigación se partió de la **hipótesis** de que el desarrollo de un sistema web que eliminará la duplicidad de información en los procesos de entrega y devolución de libros en el Vicedecanato de Economía y Administración de la Facultad 1.

Las **tareas de la investigación** que se estarán desempeñado son:

- Realizar un estudio de sistemas homólogos de forma crítica y valorativa.
- Realizar levantamiento de requisitos del cliente.
- Realizar la definición de la arquitectura y los patrones de diseño a emplear.
- Realizar el diseño de casos de prueba a realizar al sistema web.

Los **métodos científicos** empleados fueron el **método analítico sintético**, ya que se dividió la consulta de información en dos partes una primera parte dirigida a la descripción de requisitos y una segunda parte

que a su vez se dividió en otras partes dirigida a la búsqueda de información referente a la implementación de sistemas web con tecnología Symfony. El **método histórico** se utilizó para el estudio de sistemas homólogos, así como conceptos relacionados con el proceso de entrega y devolución de libros con el fin de reunir evidencia, para llegar a la comprensión y síntesis de la información con la que se está trabajando.

1 Desarrollo

1.1 Conceptos asociados

Los libros son considerados un activo importante e imprescindible en la sociedad ya que guardan en sus páginas los sucesos sobre la evolución del hombre desde sus inicios, son portadores de la historia de la humanidad, se podría decir que fueron la primera base de datos existente y que en ellos se encuentra plasmada información de toda índole. En la actualidad y según el diccionario de la lengua española se conoce como libro a un conjunto de muchas hojas de papel u otro material semejante que, encuadernadas forman un volumen (Real Academia Española, 2019), sin embargo, el primer soporte de escritura conocido es la piedra, posteriormente la arcilla, la madera, el papiro, la seda, el hueso, el bronce, la cerámica, las escamas, la palma seca, el papel, los soportes electrónicos incluso la piel humana.

Como resultado del notable desarrollo de la ciencia y la tecnología a nivel mundial, el libro electrónico se ha hecho notar en la sociedad tanto es así que la Real Academia Española ha incorporado el término en su diccionario, pero la diversidad de libros es tan grande que puede resultar difícil contarlos y clasificarlos, sin embargo, debido a los temas que abordan, a su estructura y al formato que generalmente es utilizado en su publicación se pueden clasificar en tres grandes tipos, que a su vez se clasifican en subtipos.

Dentro de los tipos de libros según el formato se encuentran los libros electrónicos o digitales y los libros de papel. Aquellos libros clasificados según la temática resguardan en su categoría a los libros científicos, literarios y especializados. Dentro del tipo de libro según la estructura se pueden encontrar los libros de textos, los libros ilustrados, los libros de referencia y los libros de gran formato (LinkedIn Learning, 2012).

Con el transcurso de los años ha mejorado notablemente la calidad de los textos, su conservación, el acceso a la información, así como la manejabilidad, el manejo de los costes y la producción. Sin duda alguna existe gran variedad de ellos y la mayoría se encuentran en bibliotecas a disposición de la población o en las múltiples instituciones educativas, por lo que se hace necesario mantener un orden, una constancia, un registro de su uso, préstamo y disponibilidad.

Existen normas de funcionamiento y reglas que se deben cumplir para solicitar un préstamo de libros en bibliotecas públicas, dichas reglas contienen información sobre el nivel del préstamo, la condición, duración y compromisos que debe hacer el lector, de forma paralela existen también sanciones vigentes que se toman en cuenta si se incurre en el incumplimiento de estas normas.

En la biblioteca universitaria de la Universidad de Gran Canaria se llevan a cabo las siguientes sanciones:

- En caso de pérdida o deterioro, la obra deberá ser repuesta y, si estuviera agotada, se deberá adquirir otra de características similares propuesta por la Biblioteca. La no reposición supondrá la

pérdida del derecho al préstamo de forma definitiva, independientemente de las medidas administrativas o legales que dispongan la legislación vigente y los reglamentos universitarios.

- Los retrasos en la devolución serán sancionados con 3 días de suspensión por cada documento y día natural de demora a partir de la fecha de devolución.
- Los retrasos en la devolución de un documento de préstamo restringido serán sancionados con 5 días de suspensión por cada documento y día natural de retraso a partir de la fecha de devolución.
- Los retrasos en la devolución de un documento de préstamo de fin de semana serán sancionados con 10 días de suspensión por cada uno de retraso. Se considerará como un día de retraso la entrega de la obra después de las 10.00 horas del primer día lectivo.
- Las sanciones serán acumulativas.
- El retraso reiterado en la devolución de las obras comportará la suspensión temporal o definitiva del préstamo. En caso de acumularse sanciones por un año en dos ocasiones, la siguiente implicará la pérdida definitiva de su condición de usuario o usuaria del servicio de préstamo de la Biblioteca Universitaria (Biblioteca Univeristaria, 2017).

En la biblioteca Prof. Don Antonio M. Grompone nombrada así en honor a su fundador y primer director del Instituto de Profesores Artigas, en Uruguay, se gestionan las sanciones de la siguiente manera:

- El incumplimiento de los plazos de devolución establecidos, será objeto de una suspensión por la misma cantidad de días de atraso.
- Por los préstamos de fin de semana, se suspenderá el préstamo a domicilio por 5 días hábiles a usuarios que no devuelvan el lunes en la mañana. Se agregarán 2 días hábiles por cada día sucesivo de atraso. La misma sanción aplicará para usuarios que usen material de sala y no devuelvan en el día (5 días hábiles).
- La acumulación de cuatro suspensiones implica la imposibilidad de hacer uso del préstamo domiciliario por el año lectivo. La entrega del/los libro/s luego de las 12 horas del lunes será considerada como un día de atraso.
- Nunca perderá su calidad de usuario.
- Las obras extraviadas o entregadas en malas condiciones deberán reponerse con el mismo documento u otro material de valor similar, en el caso de que no pudiese conseguir el mismo en un plazo de 30 días, el usuario quedará inhabilitado para hacer uso del préstamo hasta tanto no reponga el material perdido y cumpla la sanción establecida.

- Los docentes no podrán usar los servicios de préstamo hasta que no reponga o devuelva el material y cumpla la sanción establecida (Biblioteca del Instituto de Profesores "Artigas" , n.d.).

Ambas bibliotecas coinciden en varios aspectos en las respectivas sanciones por el incumplimiento de los plazos de devolución establecidos como son el establecimiento de un límite de sanciones, la suspensión temporal o definitiva del servicio de préstamo y la reposición del libro en caso de extravío del material. A pesar de que la solución que se presenta es un sistema de gestión de almacenes y no un sistema bibliotecario es importante destacar que, en cuanto al préstamo de libros, las formas de gestionar las sanciones aplicadas por pérdida de libros coinciden en su mayoría en varios aspectos con la forma de gestionar la pérdida de libros en el Vicedecanato de Economía y Administración de la Facultad 1.

En las universidades cubanas por igual se toman medidas respecto a la pérdida de libros prestados. La Universidad de Ciencias Informáticas en particular se rige por la Resolución No. 858/ 08 Procedimiento para el Control de los Medios de la Universidad donde se plantea y cito:

SEXTO: El material escolar no gastable, en caso de pérdida, inutilización o destrucción por negligencia o falta de preocupación y/o no diligencia adecuada, deberá reponerse mediante su adquisición en los establecimientos mercantiles, si está a la venta. Cuando no lo esté, el estudiante abonará la cuantía del medio de enseñanza o de vida de que se trate, de acuerdo con la escala que a continuación se expresa:

1. En cuanto a libros utilizados en la docencia, estén o no considerados textos oficiales, se atenderá a la clasificación siguiente:

1. LIBROS DE PRODUCCIÓN NACIONAL.

De Ciencias Básicas.....	\$ 10.00	por	ejemplar.
De Especialidades.....	\$ 15.00	"	"

1. LIBROS DE IMPORTACIÓN.

De Ciencias Básicas.....	\$ 15.00	por	ejemplar.
De Especialidades.....	\$ 20.00	"	"

1. EJEMPLARES SIN REPOSICIÓN.

Sean de producción nacional o de importación \$ 40.00 por ejemplar. (Se considerará un libro “de no reposición” cuando, por cualquier causa, el Centro de educación Superior no pueda adquirirlo con los recursos normales que tenga destinados a este fin) (Morell, 2008).

La Resolución No. 858/ 08, es una actualización del procedimiento para el Control de Medios para los estudiantes en la UCI. Plantea el procedimiento a llevar a cabo para mantener la debida supervisión sobre los medios de la universidad, los cuales han sido puestos a disposición de la institución y dentro de ella para estudiantes, profesores y trabajadores como propiedad social. La resolución dicta que, sin importar la categoría docente del usuario, el mismo está obligado al cuidado y preservación de la base material escolar que se tenga bajo cuidado con el fin de concientizar el respeto hacia los libros y la estabilidad económica, por lo tanto a cualquiera que incumpliese con las normativas a favor de la preservación de los libros se le aplicarán las sanciones pertinentes, las cuales están orientadas en su mayoría a la reposición del medio o al pago en cuantías en el caso de que no se encuentre a la venta, de acuerdo con la sección que le corresponda al libro en la escala presentada en la Resolución. Posterior a la realización de un análisis resulta importante resaltar que las sanciones dictadas dentro de la resolución coinciden en varios aspectos con las normativas implementadas en otras instituciones que radican dentro del país o en el exterior.

Exigir con rigor el cumplimiento de la Resolución No. 858/ 08 es una de las vías para garantizar la preservación de los libros, que es a la vez, una de las funciones de un almacén de libros, a la cual se le suman la conservación, control y expedición de mercancías y productos, además de recepción, custodia y el despacho rápido de los pedidos. A lo largo de los años, y conforme evoluciona el fenómeno logístico, el concepto de almacén ha ido variando y ampliando su ámbito de responsabilidad.

El almacén es una unidad de servicio y soporte en la estructura orgánica y funcional de una empresa comercial o industrial con objetivos bien definidos de resguardo, custodia, control y abastecimiento de materiales y productos (López, 2016). La gestión de almacenes se define como el proceso de la función logística que trata la recepción, almacenamiento y movimiento dentro de un mismo almacén hasta el punto de consumo de cualquier material – materias primas, semielaborados, terminados, así como el tratamiento e información de los datos generados (Noega Systems.Soluciones de almacenaje., 2007).

La gestión de almacenes tiene como objetivo optimizar un área logística funcional que actúa en dos etapas de flujo como lo son el abastecimiento y la distribución física para garantizar el suministro continuo y oportuno de los materiales y medios de producción requeridos para asegurar los servicios de forma ininterrumpida y rítmica (López, 2016). Un almacén de libros se gestiona de igual forma, la particularidad que los distingue es la manera de archivar los datos correspondientes a cada libro en el almacén.

1.2 Estudio de Sistemas Homólogos

A continuación, se realiza un estudio de sistemas homólogos tanto en el ámbito nacional como internacional.

1.2.1 *Estudio de sistemas homólogos en el ámbito internacional*

1. **Nombre del sistema:** PRESTO: Sistema de Préstamo de Libros-e. Guías de la BUS de la Universidad de Sevilla.

Características del sistema:

- Es un servicio de la Biblioteca de la Universidad de Sevilla que permite el préstamo de libros, audiolibros en formato digital (EPUB, PDF, Streaming) a través de internet.
- El préstamo se hará por medio de una plataforma tecnológica que permite acceder al catálogo y tomar en préstamo una serie de publicaciones en formato digital para poder leer desde el ordenador, tablets, smartphones o ereaders.
- Tiene una interfaz sencilla y fácil de usar optimizada para dispositivos Android e iOS y ordenadores con sistema operativo Windows.
- Permite tomar un préstamo de cualquier título disponible y descargarlo al dispositivo móvil de un solo clic.
- Permite navegar por el catálogo, realizar préstamos y reservas, leer online y descargar libros para leer sin conexión a internet.
- Permite la sincronización con otros dispositivos con un máximo de 6 dispositivos diferentes.
- Los requisitos previos para utilizar la aplicación son:
 - Acceso a internet.
 - Ser miembro de la comunidad universitaria (estudiante, profesor o PAS) con usuario virtual (UVU) vigente.
 - Tener instalado el programa lector de libros electrónicos: Adobe Digital Editions o la APP de ODILO (Biblioteca Universidad de Sevilla , 2019).

El sistema PRESTO: Sistema de Préstamo de Libros-e. Guías de la BUS de la Universidad de Sevilla, está pensado para publicaciones en formato digital y no para libros en formato duro como los que gestiona el Vicedecanato de Economía y Administración de la Facultad 1, principal razón por la cual se rechaza el sistema. La interfaz está diseñada para dispositivos Android, iOS y ordenadores con

sistema operativo Windows, además uno de los requerimientos para usar la aplicación es el acceso a internet, sin embargo, la solución que se brinda no requiere de acceso a internet o la instalación de un programa lector de libros electrónicos tal y como requiere la aplicación.

1.2.2 Estudio de sistemas en el ámbito nacional

- **Nombre del sistema:** Herramienta web para la gestión de la base material de estudio en la Universidad de Pinar del Río.

Características del sistema:

- La herramienta web abarca los siguientes procesos: Actualización general de los inventarios de la base material de estudio (textos, guías de estudio y otras). Gestión de las entradas y salidas de la base material de estudio entre almacenes. Gestión de préstamos y devoluciones de la base material de estudio a estudiantes y profesores.
- La herramienta permite crear un inventario general actualizado de la base material de estudio y su estado real en todas las áreas de la universidad.
- La herramienta posibilita conocer la existencia y ubicación de la base material de estudio según plan bibliográfico para cada carrera, años y semestre lo que permite gestionar los movimientos e intercambios necesarios de la misma.
- La herramienta tiene como característica que permite conocer los movimientos de entradas y salidas de la base material de estudio entre almacenes.
- Permite gestionar los préstamos y devoluciones de la base material de estudio a estudiantes y profesores.
- La herramienta permite conocer el déficit de base material de estudio que permita establecer una estrategia racional para su aseguramiento mediante la elaboración de materiales y otras acciones.
- Permite conocer el exceso de base material de estudio lo que permite desarrollar un plan de acciones encaminadas a la disminución de inventarios en almacenes mediante el intercambio entre almacenes y con otras universidades (Carmona, 2013).

A pesar de que la herramienta web para la gestión de la base material de estudio en la Universidad de Pinar del Río, cuenta con un módulo para gestionar el préstamo para estudiantes y profesores y un módulo para generar un inventario actualizado sobre la base material de estudio, se descarta la herramienta, ya que no realiza reporte por pérdida de libros para estudiantes y profesores, no cuenta con un módulo para realizar el reporte trimestral del estado de extracción de los libros, además la herramienta

no genera un Modelo de entrega y devolución de libros, la cual es la ficha que utiliza el Vicedecanato de Economía y Administración de la Facultad 1 para mantener un control sobre el proceso de entrega y devolución de libros para cada estudiante y profesor.

- **Nombre del sistema:** Informatización del servicio de préstamo de literatura docente.

Características del sistema:

- Es una aplicación informática que permite el control del proceso de préstamo de literatura docente en la Universidad de Ciencias Médicas de Pinar del Río.
- El sistema realiza los siguientes procesos: Solicitar Servicio. Controlar préstamo de literatura docente. Actualizar listado de estudiantes. Controlar literatura docente. Gestionar usuario.
- El sistema permite mantener una mejor organización y control de la literatura docente disponible para préstamo a los usuarios de las diferentes carreras.
- Permite el registrar los datos personales de los usuarios y la literatura utilizada (Informatización del servicio de préstamo de literatura docente, 2017).

El sistema tiene como objetivo el control de préstamo de la literatura docente en la Universidad de Ciencias Médicas de Pinar del Río, pero no se encarga de gestionar el inventario de los libros con los que cuenta la universidad, no realiza reportes por pérdida de libros para estudiantes y profesores y no genera un reporte trimestral para mantener un control sobre el estado de extracción de los libros.

- **Nombre del sistema:** ABCD.

Características del sistema:

En entrevista realizada al Director del Centro de Informatización de la Gestión Documental CIGED, se expresó que el ABCD es un sistema de gestión bibliotecaria, que ofrece herramientas para la gestión de bases de datos, entrada de datos, estadísticas, la circulación, control de publicaciones periódicas y funciones de búsqueda. El sistema trabaja con permisos a nivel de usuario, por lo tanto, todos los usuarios no tienen acceso a las mismas funcionalidades.

- Tecnologías sobre las que se desarrolló el sistema:
 - UML 2.0, Java, PostgreSQL 9.3, RAP 3.0M5.

La principal razón por la cual se rechaza el sistema ABCD es que es un sistema de gestión bibliotecaria, lo cual significa que a pesar de que contiene módulos para realizar préstamos y generar reportes por pérdida de libros, el sistema utiliza una forma de catalogación de libros que no coincide con la manera de gestionar un almacén, además las funcionalidades se trabajan como procesos, lo cual imposibilita habilitar y deshabilitar módulos siendo este último uno de las peticiones del Vicedecanato de Economía y Administración.

Tabla 1. Tabla de comparación entre sistemas.

Nombre del sistema	Creación de Módulo de entrega y recepción de libros	Creación de inventario	Realización de reporte trimestral	Realización de reportes por pérdida de libros
PRESTO	No	No	No	No
Herramienta web para la gestión de la base material de estudio en la Universidad de Pinar del Río.	No	Si	No	No
Informatización del servicio de préstamo de literatura docente.	No	No	No	No
ABCD	No	No	No	Si

Las características de los sistemas previamente analizados, en comparación a la solución que se desea brindar como resultado de este trabajo no satisfacen las necesidades y requerimientos del Vicedecanato de Economía y Administración de la Facultad 1. Los sistemas analizados no se encargan de realizar la gestión de un Modelo por pérdida de libros para cada estudiante o profesor que solicite un préstamo, los sistemas en su mayoría no se encargan de realizar la creación de un inventario para mantener un control actualizado en el Vicedecanato de Economía y Administración de la Facultad 1. Además, no tienen implementada una funcionalidad que permita realizar un reporte trimestral para conocer el estado de

extracción de los libros en la Facultad 1 y a excepción del Sistema de Gestión Bibliotecaria ABCD no se encargan de realizar un reporte por pérdida de libros para cada estudiante o profesor que así lo requiera, por lo que se decide desarrollar una aplicación web que responda a las necesidades del Vicedecanato de Economía y Administración de la Facultad 1.

1.3 Ambiente de desarrollo

Para dar solución al desarrollo de la aplicación web se hace una valoración crítica de los lenguajes y el conjunto de herramientas y tecnologías a utilizar, guiados por una metodología de desarrollo.

1.3.1 Metodología de desarrollo

La metodología RUP es básicamente una guía para usar efectivamente el Lenguaje Unificado de Modelación (Unified Modeling Language –UML), el cual es un lenguaje estándar que permite expresar claramente requerimientos, arquitectura y diseños. Para es el desarrollo de la investigación se selecciona RUP como metodología de desarrollo porque permite una evaluación de los avances del proyecto por su gestión debido a que basa en el modelo incremental o evolutivo el cual se realiza por pequeños incrementos y en cada uno se puede refinar el anterior, permitiendo encontrar errores en la aplicación. RUP es preferentemente aplicable a grandes equipos de desarrollo y a grandes proyectos y tiene como desventaja que la detección de requisitos tardíamente tiene una recuperación costosa, sin embargo, es una metodología personalizable lo que hace posible su adaptación a proyectos de cualquier escala, incluso para equipos pequeños de hasta 1 persona. Contiene un “Kit de Desarrollo” que provee soporte para configurar el proceso para adaptarse a las necesidades del equipo de desarrollo que la use, además asegura la producción de software de alta calidad, el cual llega a las necesidades del cliente final, por medio de un horario y presupuesto predecibles, antes de someter grandes cantidades de recursos (Jaramillo, 2016).

1.3.2 Herramientas

- **Symfony v.4.0**

Es un framework diseñado para desarrollar aplicaciones web basado en el patrón Modelo Vista Controlador. Además de automatizar las tareas, proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja y está desarrollado completamente en PHP 5.3. Es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y Microsoft SQL Server, tiene un sistema operativo multiplataforma ya que se puede ejecutar tanto en plataformas *nix (por ejemplo, Unix, Linux) como en plataformas Windows. Se seleccionó symfony como lenguaje de programación porque es un framework automatizado lo cual implica que contiene

instrucciones para integrar paquetes, posibilita al desarrollador trabajar con menos código y su optimización elimina cualquier dependencia no esencial (SensioLabs, 2017).

- **PHP v.7.0**

Es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML por lo que permite configurar el servidor web para que procese todos los ficheros HTML. PHP es el lenguaje de programación para el desarrollo del backend más popular y uno de los más utilizados a nivel global ya que con este se pueden desarrollar sitios y aplicaciones de todo tipo. Durante esta investigación se estará trabajando con PHP porque es un lenguaje libre y abierto lo cual lo hace la opción idónea para desarrollar un sistema web, dispone además de paquetes autoinstalables que pueden ser utilizados tanto en sistema operativos *nix como en Windows (php.net, 2017).

- **Servidor web Apache**

Apache es un servidor web con soporte para HTTP 1.1 y para HTTP2 según la norma RFC 7540. Tiene soporte para VirtualHost tanto para dominios como para IP respetando la normal norma RFC 2616, lo que permite alojar varios dominios en un mismo servidor con una misma IP. Es un servidor gratuito y de código abierto. Es un servidor altamente extensible y adaptable mediante módulos. Tiene funciones incorporadas para autenticación y validación de usuarios, además tiene soporte para lenguajes como Perl, PHP y Python (León, 2019).

- **Brackets v.1.12**

Es un editor de código HTML, CSS y JavaScript desarrollado por Adobe que, ofrece herramientas para agilizar el proceso de creación de páginas y mejorar la conversión entre el editor gráfico y su interpretación a través del lenguaje utilizado. Se trata de una herramienta Open Source con una interfaz clara y directa que dispone de detección automática del código y ayudas en la escritura, sangriado y coloración para identificar segmentos, pudiendo organizar todos los elementos de un proyecto a través del área, en la que se integran los archivos de código y los propios elementos visuales que formen parte de la web que se está creando. Para el desarrollo de esta investigación se estará trabajando con Brackets porque una de sus ventajas es brindar la posibilidad de abrir el proyecto que se esté desarrollando en el navegador y ver en tiempo real las modificaciones realizadas al código fuente, una característica importante a considerar en caso de que surjan nuevos requisitos por parte del cliente durante el período de implementación (Adobe, 2019).

- **Navegador web Firefox**

Firefox es un navegador y software de código abierto gratuito que se caracteriza por un gran número de posibles personalizaciones y ajustes. Tiene como principales características que es multiplataforma ya que existen versiones de Mozilla Firefox para Windows, Linux y Mac. Permite abrir simultáneamente varias páginas web de tal forma que cada una se visualiza en una pestaña independiente, posibilita además restaurar sesión. Firefox ofrece más de 1000 complementos que permiten aumentar las prestaciones por defecto de este navegador web: lectura de noticias RSS, herramientas web y de desarrollo, descargas de archivos, privacidad y seguridad, herramientas de búsqueda, marcadores, diccionarios, multimedia, entre otros. El uso de un administrador de complementos facilita las operaciones de instalación, desinstalación y desactivación. Para el desarrollo de esta investigación se estará trabajando con el navegador web Firefox ya que brinda gran accesibilidad y dispone de plugins necesarios para visualizar todo tipo de contenido web y multimedia (Instituto Superior de Formación y Recursos en Red para el Profesorado, 2008).

- **Visual Paradigm for UML v.8.0**

Es un editor de diagramas simple e intuitivo que posee facilidades como arrastrar para crear una forma, guías de alineación que ayudan a colocar formas con precisión, Paradigm soporta una amplia gama de necesidades de visualización, desde diseño de software, modelado de datos, mapeo de procesos de negocio, análisis estratégico, mapeo mental y programación de proyectos, y es ampliamente adoptado en diferentes sectores como negocios, educación y unidades sociales. Se seleccionó Visual Paradigm para el desarrollo de esta investigación porque es una herramienta multiplataforma, que a través de la representación mediante diagramas es capaz de representar el ciclo de vida completo del sistema (Visual Paradigm, 2010).

- **PostgreSQL v.9.4**

PostgreSQL es un potente sistema de base de datos relacional de objetos de código abierto que usa y amplía el lenguaje SQL combinado con características que almacenan y escalan de manera segura las cargas de trabajo de datos más complicadas. El sistema se ha ganado una sólida reputación por su arquitectura comprobada, confiabilidad, integridad de datos, conjunto de características robustas, extensibilidad y la dedicación de la comunidad de código abierto detrás del software para ofrecer soluciones innovadoras y de alto rendimiento. Se ejecuta en todos los principales sistemas operativos, tiene complementos potentes como el extensor de base de datos geoespaciales PostGIS. Se ha seleccionado PostgreSQL como gestor de base de datos porque contiene características que proporcionan el soporte necesario para desarrollar una aplicación, prioriza la protección de la integridad de los datos y crea entornos tolerantes a fallas, además de que posibilita administrar los datos sin importar cuán grande o pequeño sea el conjunto de datos (The PostgreSQL Global Development Group, 2010)

2 Propuesta de solución

Se realiza una propuesta de solución que estará guiada por la metodología RUP, la cual, en cada una de sus fases genera una serie de artefactos para mantener de forma organizada la estructura del sistema. Como resultado de la investigación, la propuesta de solución se verá representada en un sistema web a través del cual el cliente será capaz de gestionar el almacén de libros de la Facultad 1 de forma automatizada respondiendo a las necesidades del Vicedecanato de Economía y Administración. La propuesta de solución resuelve aspectos como son la creación y actualización de inventarios, reportes trimestrales, controles de entrega y devolución de libros y modelos por pérdida de libros de forma digital y a su vez permitirá conocer el estado de extracción de los libros en tiempo real. La principal ventaja que ofrecerá el sistema será mantener un control actualizado sobre el Control de Entrega y Recepción de Libros de cada estudiante o trabajador que solicite un préstamo o realice una devolución en la Facultad 1. El objetivo principal de la propuesta de solución será eliminar la duplicidad de la información, para ello a cada uno de estos documentos se les asignará un identificador que se registrará de forma única.

2.1.1 Diagrama de casos de uso del negocio

A continuación, se presentan los diagramas de caso de uso del negocio que muestran los principales actores del negocio y representan el proceder de una serie de acciones que producen un resultado de valor para el actor del negocio. En el diagrama se muestran dos actores del negocio. El actor Vicedecana se encarga de realizar los procesos Elaborar Inventario. Realizar entrega de libros. Realizar devolución de libros. Elaborar reporte por pérdida de libros. El actor Persona, que puede ser un estudiante o un profesor, es quien realiza la solicitud de préstamo o devolución de libros.

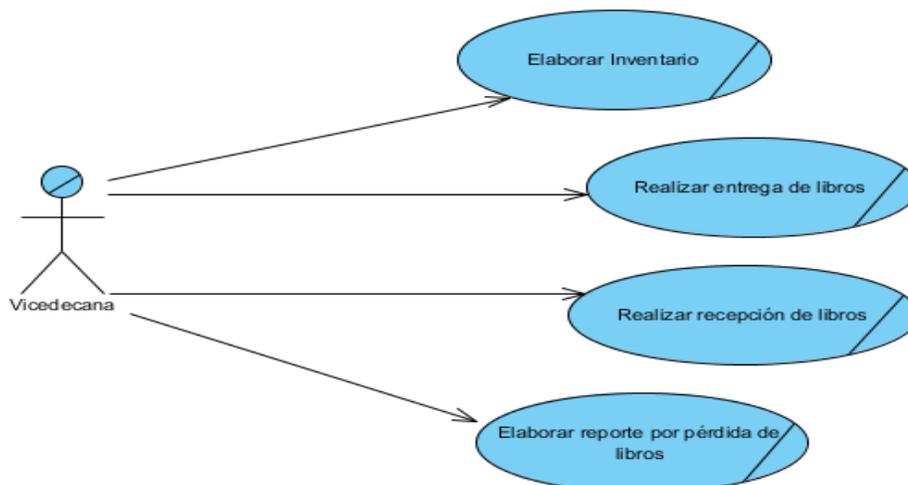


Fig. 1. Diagrama de caso de uso del negocio.

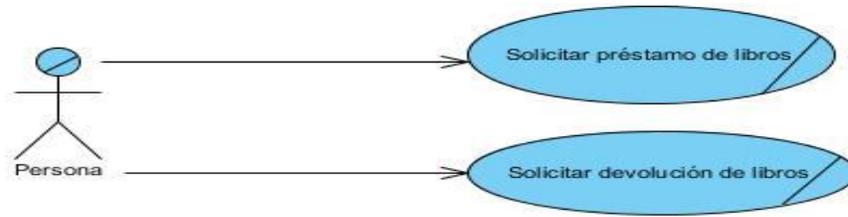


Fig. 2. Diagrama de caso de uso del negocio.

2.1.2 Diagrama de actividades

A continuación, se presentan los diagramas de actividades, que muestran el flujo de actividades ejecutadas por la propuesta de solución como sistema.

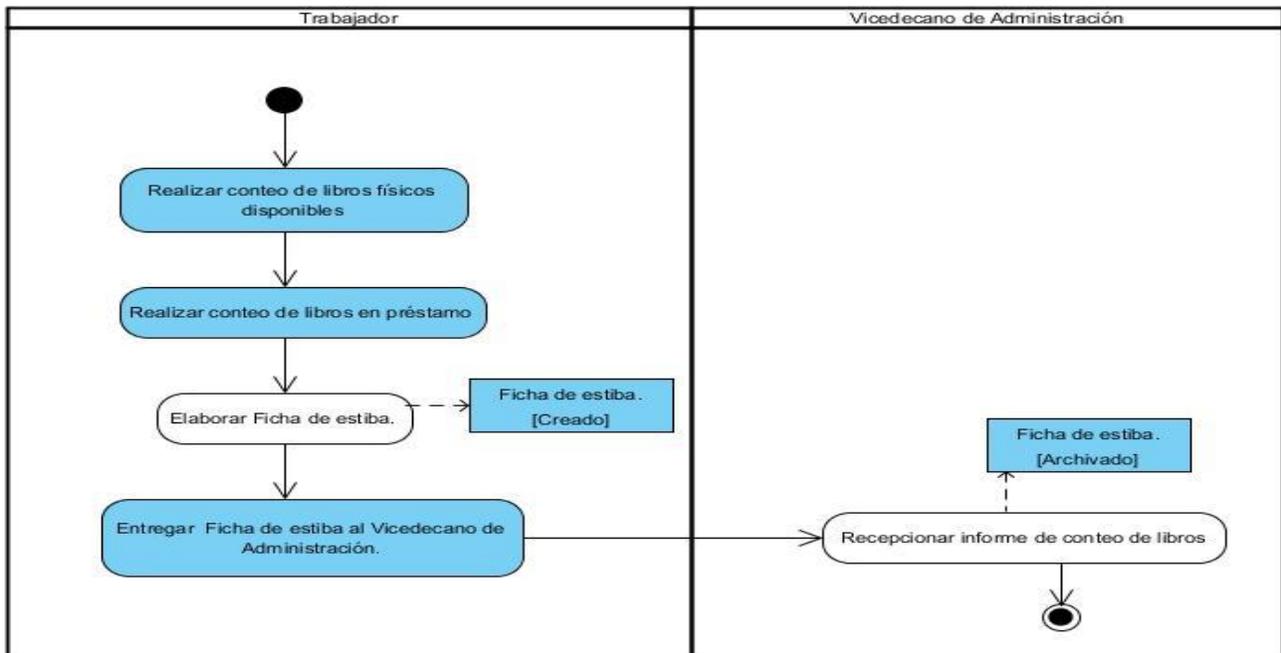


Figura 3. Diagrama de actividades del caso de uso "Elaborar Inventario".

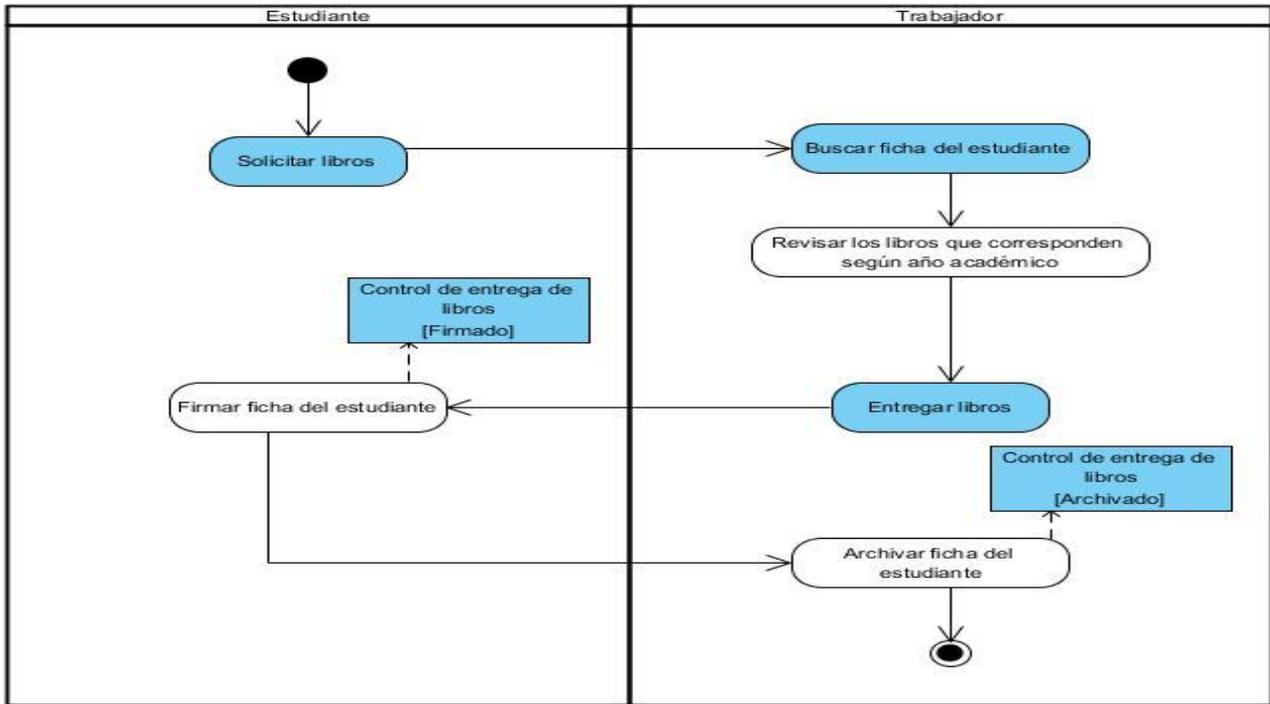


Figura 4. Diagrama de actividades del caso de uso "Solicitar préstamo".

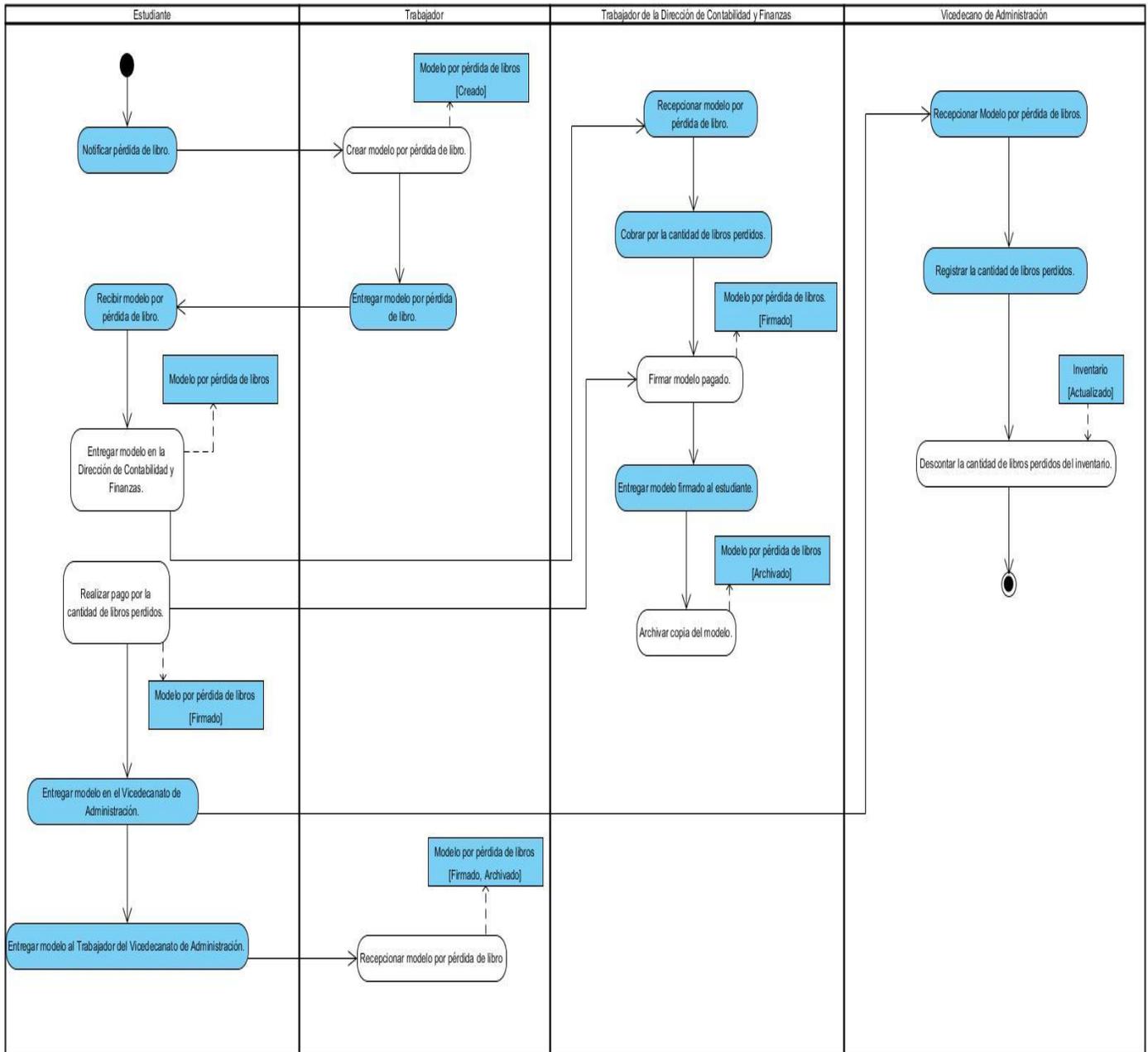


Figura 5. Diagrama de actividades del caso de uso "Elaborar reporte por pérdida de libros".

2.2 Especificación de requisitos

Los requisitos funcionales son declaraciones de los servicios que prestará el sistema, en respuesta a las solicitudes del cliente final, para el desarrollo de esta investigación se recolectó un total de 18 requisitos funcionales, a continuación se presenta la tabla de requisitos funcionales que responden a las necesidades del Vicedecanato de Economía y Administración de la Facultad 1.

Tabla 2. Tabla de Requisitos funcionales.

No.	Nombre del requisito.
RF. 1	Autenticar Usuario.
RF. 2	Adicionar control de entrega de libros.
RF. 3	Modificar control de entrega de libros.
RF. 4	Listar control de entrega de libros.
RF. 5	Eliminar control de entrega de libros
RF. 6	Adicionar inventario.
RF. 7	Modificar inventario.
RF. 8	Listar inventario.
RF. 9	Eliminar inventario.
RF. 10	Adicionar modelo por pérdida de libros.
RF. 11	Modificar modelo por pérdida de libros.
RF. 12	Eliminar modelo por pérdida de libros.
RF. 13	Listar modelo por pérdida de libros.
RF. 14	Adicionar reportes.
RF. 15	Modificar reportes.
RF. 16	Listar reportes.
RF. 17	Eliminar reportes.
RF. 18	Adicionar Buscar.

A continuación, se presenta el documento visión como artefacto entregable que recoge la visión general de requerimientos, características claves y las restricciones principales de los requisitos funcionales. Para otorgarle criterios de complejidad y prioridad a los requisitos recolectados se partió de que el objetivo de RUP es priorizar aquellas tareas del proyecto que sean más importantes para el cliente a nivel de negocio,

realizar tareas completas en cuanto funcionalidad, mostrar el resultado al cliente y, una vez obtenido un feedback, mejorar todo lo necesario. Para ambos criterios se trabajará con 3 clasificaciones: Alta, Media y Baja, clasificación que se le será otorgada al requisito de acuerdo a la importancia de la priorización de aquellas tareas o funcionalidades que impliquen valor de negocio.

Tabla 3. Tabla del documento visión.

Nº	Nombre	Descripción	Complejidad	Prioridad para cliente
RF. 1	Autenticar usuario	El usuario tiene los permisos necesarios para autenticarse en el sistema.	Baja	Media
RF. 2	Adicionar Modelo de entrega y devolución de libros.	El usuario adiciona al sistema un “ Modelo de entrega y devolución de libros ” que contiene los campos: No., Nombre y apellidos, No. Solapín, Año, Grupo, Título, Fecha entrega, Fecha devolución y Observación.	Alta	Alta
RF. 3	Modificar Modelo de entrega y devolución de libros .	El usuario modifica los campos del Modelo de entrega y devolución de libros : fecha de devolución, observación y título.	Alta	Alta
RF. 4	Listar Modelo de entrega y devolución de libros .	Se listan los Modelo de entrega y devolución de libros adicionados al sistema mostrando los siguientes campos: No., Nombre y apellidos, No. Solapín, Año, Grupo, Título, Fecha entrega, Fecha devolución y Observación.	Media	Alta

RF. 5	Eliminar Modelo de entrega y devolución de libros .	El usuario elimina un Modelo de entrega y devolución de libros de los que se encuentran registrados en el sistema.	Alta	Alta
RF. 6	Adicionar inventario.	El usuario adiciona al sistema un nuevo inventario con los siguientes campos: No., fecha, cantidad de libros disponibles, cantidad de libros en préstamo, títulos disponibles, total.	Alta	Alta
RF. 7	Modificar Inventario	El usuario modifica los campos del inventario: Cantidad de libros disponibles, cantidad de libros en préstamo, títulos disponibles y total.	Alta	Alta
RF. 8	Listar Inventario	Se listan los inventarios adicionados al sistema mostrando los siguientes campos: No., fecha, cantidad de libros disponibles, cantidad de libros en préstamo Títulos disponibles, Total.	Media	Alta
RF. 9	Eliminar inventario	El usuario elimina un inventario de los que se encuentran registrados en el sistema.	Alta	Alta
RF. 10	Adicionar Modelo por pérdida de libros	El usuario adiciona un Modelo por pérdida de libros que contiene los campos: Nombre del estudiante, apellidos, descripción, importe.	Alta	Alta

RF. 11	Modificar modelo por pérdida de libros	El usuario modifica los campos del Modelo por pérdida de libros: Nombre y apellidos, descripción e importe.	Alta	Alta
RF. 12	Listar Modelo por pérdida de libros	Se listan los Modelos por pérdida de libros registrados en el sistema mostrando los siguientes campos: nombre del estudiante, apellidos, descripción, importe.	Media	Alta
RF. 13	Eliminar Modelo por pérdida de libros.	Se elimina uno de los modelos por perdida de libros registrados en el sistema.	Media	Alta
RF. 14	Adicionar Reporte trimestral.	El usuario crea un reporte trimestral donde se registran los siguientes datos: Cantidad de estudiantes que no han sacado libros, Cantidad de estudiantes por año que han sacado libros, Libros nunca que se han sacado, Libros que más se solicitan, Año académico, Matrícula.	Alta	Alta
RF. 15	Modificar Reporte trimestral	El usuario selecciona un reporte trimestral de los registrados en el sistema permitiendo modificar todos los campos.	Alta	Media
RF. 16	Listar Reporte trimestral	Se listan los reportes trimestrales registrados en el sistema mostrando los campos: Cantidad de estudiantes que no han sacado libros,	Baja	Media

		Cantidad de estudiantes por año que han sacado libros, Libros nunca que se han sacado, Libros que más se solicitan, Año académico, Matrícula.		
RF. 17	Eliminar Reporte trimestral	Se elimina un reporte trimestral de los eliminados en el sistema.	Media	Alta
RF. 18	Adicionar Buscar	El usuario introduce valores en algunos de los siguientes criterios: año académico, nombre, grupo en el caso del estudiante y área de trabajo en el caso del trabajador y presiona el botón Buscar.	Alta	Alta

2.3 Requisitos no funcionales

Los requisitos no funcionales responden a las características que tendrá el sistema que se implementará de acuerdo a las solicitudes del Vicedecanato de Economía y Administración de la Facultad 1.

Apariencia o interfaz:

1. En la interfaz debe predominar los colores blanco y negro y debe tener el logo UCI.

Eficiencia:

2. El tiempo de respuesta del sistema debe ser menor de cinco segundos.

Usabilidad:

3. El usuario debe estar autenticado en el sistema.
4. Facilidad de uso por parte de los usuarios: el sistema debe presentar una interfaz amigable que permita la fácil interacción con el mismo y llegar de manera rápida y efectiva a la información buscada.

Portabilidad:

5. El sistema debe permitir su uso en diferentes sistemas operativos, como Linux y Windows.

Seguridad:

6. El sistema debe garantizar seguridad mediante el protocolo https y políticas de soberanía tecnológica.

Desarrollo de software:

7. El sistema se debe implementar mediante una aplicación web.
8. El sistema debe tener comunicación con la base de datos.

2.3.1 Diagrama de caso de uso del sistema

A continuación, se presenta el diagrama de caso de uso del sistema que representa la relación entre los actores y los casos de uso de la propuesta de solución y muestra el comportamiento de la propuesta de solución como sistema mediante su interacción con los usuarios.

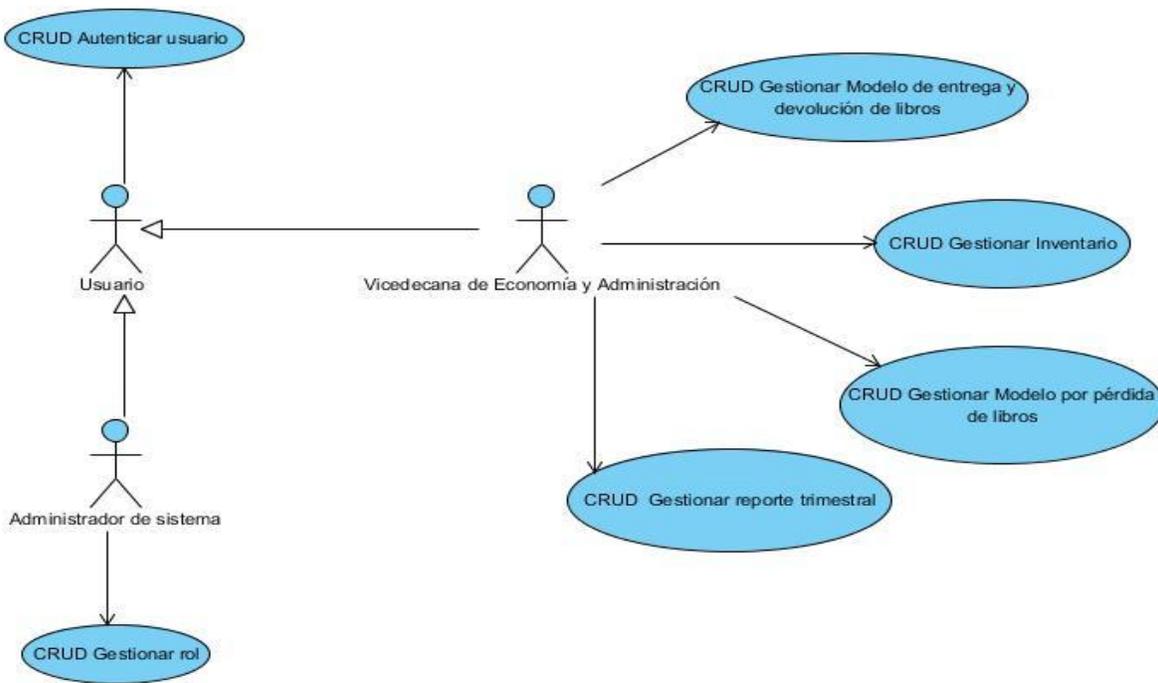


Figura 6. Diagrama de caso de uso del sistema.

A continuación se presenta el desarrollo de flujos de casos de uso de la propuesta de solución como sistema.

Tabla 4. Caso de Uso “Gestionar control de entrega de libros”.

Objetivo	Permitir Gestionar el Modelo de entrega y devolución de libros.
Actores	Vicedecana de Economía y Administración
Resumen	El usuario gestiona el Modelo de entrega y devolución de libros.
Complejidad	Alta
Prioridad	Crítico
Precondiciones	El usuario se ha autenticado en el sistema y cuenta con los permisos necesarios para ejecutar esta acción.

Postcondiciones	Se registró, eliminó, modificó y listó el Modelo de entrega y devolución de libros.	
Flujo de eventos		
Flujo básico Gestionar Modelo de entrega y devolución de libros.		
	Actor	Sistema
1.	Solicita adicionar un Modelo de entrega y devolución de libros desde la interfaz principal.	
2.		<p>Muestra alguna de las siguientes acciones:</p> <p>a) Si decide crear un nuevo Modelo de entrega y devolución de libros, ir a la sección 1 "Crear Modelo de entrega y devolución de libros".</p> <p>b) Si decide modificar el Modelo de entrega y devolución de libros, ir a la sección 2 "Modificar datos de Modelo de entrega y devolución de libros".</p> <p>c) Si decide eliminar un Modelo de entrega y devolución de libros, ir a la sección 3 "Eliminar Modelo de entrega y devolución de libros".</p> <p>d) Si decide listar un Modelo de entrega y devolución de libros, ir a la sección 4 "Listar Modelo de entrega y devolución de libros".</p>
Sección 1: "Crear Modelo de entrega y devolución de libros"		
Flujo básico Crear Modelo de entrega y devolución de libros.		
	Actor	Sistema
1.	Solicita crear un Modelo de entrega y devolución de libros.	Muestra la interfaz correspondiente para "Crear Modelo de entrega y devolución de libros".
2.	<p>Inserta los datos del nuevo Modelo de entrega y devolución de libros:</p> <ul style="list-style-type: none"> • Nombre y apellidos • Solapín • Año • Grupo • Título • Fecha entrega • Fecha devolución • Observación 	
3.		Valida los datos. En caso de ser incorrectos los datos introducidos, ver Flujo Alterno 1.a Datos incorrectos.
4.		Muestra el mensaje "Modelo de entrega y devolución de libros creado correctamente" y termina el caso de uso.
Flujos alternos		
Nº 1.a Datos incorrectos		
	Actor	Sistema
1.		Muestra el mensaje de error.
2.		Vuelve al paso 2 del flujo básico "Modelo de entrega y devolución de libros".

Sección 2: “Modificar datos de Modelo de entrega y devolución de libros”		
Flujo básico Modificar datos de Modelo de entrega y devolución de libros.		
	Actor	Sistema
1.	Selecciona la opción de modificar los datos de un Modelo de entrega y devolución de libros.	
2.		Muestra el listado de Modelo de entrega y devolución de libros que existen en el sistema.
3.	Selecciona el Modelo de entrega y devolución de libros que se desee modificar los datos.	
4.		Obtiene de la base de datos, la información del Modelo de entrega y devolución de libros existente donde se incluye su Nombre, apellidos, año, grupo, título fecha entrega, fecha devolución.
5.	Inserta los datos que desea modificar del Modelo de entrega y devolución de libros: <ul style="list-style-type: none"> • Grupo • Título • Fecha entrega • Fecha devolución • Observación 	
6.		Valida que los datos introducidos fueron correctos. En el caso que los datos insertados sean incorrectos, ver flujo Alternativo 2.a Campos Incorrectos.
7.		Modifica los datos del Modelo de entrega y devolución de libros en la base de datos y muestra el mensaje “Los datos del Modelo de entrega de libros han sido modificados correctamente” y termina el caso de uso.
Flujos alternos		
Flujo Alternativo 2.a Campos Incorrectos		
	Actor	Sistema
1.		Muestra un mensaje de error “No se pudo modificar con éxito el Modelo de entrega y devolución de libros pues uno de sus datos es incorrecto”.
2.		Vuelve al paso 5 del Flujo Básico “Modificar datos de Modelo de entrega y devolución de libros”.
Sección 3: “Eliminar Modelo de entrega y devolución de libros”		
Flujo básico Eliminar modelo de entrega y devolución de libros		
	Actor	Sistema
1.	Selecciona la opción de eliminar un	

	Modelo de entrega y devolución de libros.	
2.		Muestra un listado con todos los Modelos de entrega y devolución de libros existentes en la base de datos.
3.	Selecciona el Modelo de entrega y devolución de libros que desee eliminar.	
4.		El Modelo de entrega y devolución de libros es eliminado con éxito y se muestra el mensaje : "Modelo de entrega y devolución de libros eliminado satisfactoriamente" termina el caso de uso.
Sección 4: "Listar Modelo de entrega y devolución de libros"		
Flujo básico Listar Modelo de entrega y devolución de libros		
	Actor	Sistema
1.	Selecciona la opción listar Modelo de entrega y devolución de libros.	
2.		Muestra un listado con todos los Modelo de entrega y devolución de libros existentes en la base de datos donde se incluye por cada uno su Nombre, apellidos, año, grupo, título, No. Solapín y termina el caso de uso.
Relaciones	CU Incluidos	N/A
	CU Extendidos	N/A
Requisitos no funcionales	Deben predominar los colores blanco y negro.	
Referencias	N/A	
Asuntos pendientes	N/A	
Prototipo de interfaz gráfica de usuario		

Tabla 5. Caso de Uso “Gestionar inventario”.

Objetivo	Permitir Gestionar el inventario.	
Actores	Vicedecana de Economía y Administración.	
Resumen	El usuario gestiona los inventarios.	
Complejidad	Alta	
Prioridad	Crítico	
Precondiciones	El usuario se ha autenticado en el sistema y cuenta con los permisos necesarios para ejecutar esta acción.	
Postcondiciones	Se registró, eliminó, modificó y listó los controles de entrega de libros.	
Flujo de eventos		
Flujo básico Gestionar inventario		
	Actor	Sistema
1.	Solicita adicionar inventario desde la interfaz principal.	
2.		Muestra alguna de las siguientes acciones: a) Si decide crear un nuevo inventario, ir a la sección 1 “Crear inventario”. b) Si decide modificarlos datos de un inventario, ir a la sección 2“Modificar datos de inventario”. c) Si decide eliminar los datos del inventario, ir a la sección 3 "Eliminar inventario". d) Si decide listar los datos de inventarios, ir a la sección 4 “Listar inventarios”.
Sección 1: “Crear inventario”		

Flujo básico Crear inventario		
	Actor	Sistema
1.	Solicita crear un inventario.	
2.		Muestra la interfaz correspondiente para "Crear inventario".
3.	Inserta los datos del nuevo inventario: <ul style="list-style-type: none"> • No. • Fecha • Cantidad de libros disponibles • Cantidad de libros en préstamo • Títulos disponibles • Total 	
4.		Valida los datos. En caso de ser incorrectos los datos introducidos, ver Flujo Alterno 1.a Datos incorrectos.
5.		Muestra el mensaje "Inventario creado correctamente" y termina el caso de uso.
Flujos alternos		
Nº 1.a Datos incorrectos		
	Actor	Sistema
1.		Muestra el mensaje de error.
2.		Vuelve al paso 2 del flujo básico "Crear inventario".
Sección 2: "Modificar datos del inventario"		
Flujo básico Modificar datos del inventario		
	Actor	Sistema
1.	Selecciona la opción de modificar los datos de un inventario.	
2.		Muestra el listado de inventarios existentes en el sistema.
3.	Selecciona el inventario que se desee modificar los datos.	
4.		Obtiene de la base de datos, la información del inventario existente donde se incluye su No., fecha, cantidad de libros disponibles, cantidad de libros en préstamo, títulos disponibles, total.
5.	Inserta los datos que desea modificar del inventario: <ul style="list-style-type: none"> • Fecha • Cantidad de libros disponibles • Cantidad de libros en préstamo • Títulos disponibles • Total 	
6.		Valida que los datos introducidos fueron correctos. En el caso que los datos insertados sean incorrectos, ver flujo Alterno 2.a Campos Incorrectos.
7.		Modifica los datos del inventario en la base de datos y muestra el mensaje "Los datos del

		inventario han sido modificados correctamente” y termina el caso de uso.
Flujos alternos		
Flujo Alterno 2.a Campos Incorrectos		
	Actor	Sistema
1.		Muestra un mensaje de error “No se pudo modificar con éxito el inventario pues uno de sus datos es incorrecto”.
2.		Vuelve al paso 5 del Flujo Básico “Modificar datos del inventario”.
Sección 4: “Listar inventarios”		
Flujo básico Listar inventarios		
	Actor	Sistema
1.	Selecciona la opción de eliminar un inventario.	
2.		Muestra un listado con todos los inventarios existentes en la base de datos.
3.	Selecciona el inventario que desee eliminar.	
4.		El inventario es eliminado con éxito y se muestra el mensaje : “Inventario eliminado satisfactoriamente” termina el caso de uso.
Relaciones	CU Incluidos	N/A
	CU Extendidos	N/A
Requisitos no funcionales	Deben predominar los colores blanco y negro.	
Referencias	N/A	
Asuntos pendientes	N/A	
Prototipo de interfaz gráfica de usuario		

Tabla 6. Caso de Uso "Gestionar reporte por pérdida de libros".

Objetivo	Permitir Gestionar el Reporte por pérdida de libros.	
Actores	Vicedecana de Economía y Administración.	
Resumen	La Vicedecana de Economía y Administración gestiona Modelo por pérdida de libros.	
Complejidad	Alta	
Prioridad	Crítico	
Precondiciones	El usuario está autenticado, y cuenta con los permisos necesarios para ejecutar esta acción.	
Postcondiciones	Se registró, eliminó, modificó y listó el Reporte por pérdida de libros satisfactoriamente.	
Flujo de eventos		
Flujo Gestionar reporte por pérdida de libros.		
	Actor	Sistema
1.	Solicita adicionar por Reporte pérdida de libros desde la interfaz principal.	
2.		<p>Muestra alguna de las siguientes acciones:</p> <p>a) Si decide crear un nuevo Reporte por pérdida de libros, ir a la sección 1 "Crear Reporte por pérdida de libros".</p> <p>b) Si decide modificarlos datos de un Reporte por pérdida de libros, ir a la sección 2 "Modificar datos de Reporte por pérdida de libros".</p> <p>c) Si decide eliminar los datos del Reporte por pérdida de libros, ir a la sección 3 "Eliminar Reporte por pérdida de libros".</p> <p>d) Si decide listar los datos de los Reporte por pérdida de libros, ir a la sección Reporte por pérdida de libros".</p>
Sección 1: "Crear Reporte por pérdida de libros"		
Flujo básico Crear Reporte por pérdida de libros.		

	Actor	Sistema
1.	Solicita crear un Reporte por pérdida de libros.	
2.		Muestra la interfaz correspondiente para “Crear un Reporte por pérdida de libros”.
3.	Inserta los datos del nuevo Reporte por pérdida de libros: <ul style="list-style-type: none"> • Nombre y apellidos de la persona • Nombre y apellidos del trabajador • Descripción • Importe • Fecha • Firma 	
4.		Valida los datos. En caso de ser incorrectos los datos introducidos, ver Flujo Alterno 1.a Datos incorrectos.
5.		Muestra el mensaje “Reporte por pérdida de libros creado correctamente” y termina el caso de uso.

Flujos alternos

Nº 1.a Datos incorrectos

	Actor	Sistema
1.		Muestra el mensaje de error.
2.		Vuelve al paso 2 del flujo básico “Crear Reporte por pérdida de libros”.

Sección 2: “Modificar datos de Reporte por pérdida de libros”

Flujo básico Modificar datos de Reporte por pérdida de libros.

	Actor	Sistema
1.	Selecciona la opción de modificar Reporte por pérdida de libros.	
2.		Muestra el listado de Reporte por pérdida de libros existentes en el sistema.
3.	Selecciona el Reporte por pérdida de libros del sistema que se desee modificar los datos.	
4.		Obtiene de la base de datos, la información del Reporte por pérdida de libros existente donde se incluye su Nombre y apellidos del estudiante, descripción, importe.
5.	Inserta los datos que desea modificar del Reporte: <ul style="list-style-type: none"> • Descripción • Importe 	
6.		Valida que los datos introducidos fueron correctos. En el caso que los datos insertados sean incorrectos, ver flujo Alterno 2.a Campos Incorrectos.
7.		Modifica los datos del Reporte por pérdida de libros en la base de datos y muestra el mensaje

		"Los datos del Reporte por pérdida de libros han sido modificados correctamente" y termina el caso de uso.	
Flujos alternos			
Flujo Alterno 2.a Campos Incorrectos			
	Actor	Sistema	
1.		Muestra un mensaje de error "No se pudo modificar con éxito al Reporte por pérdida de libros pues uno de sus datos es incorrecto".	
2.		Vuelve al paso 5 del Flujo Básico "Modificar Reporte por pérdida de libros".	
Sección 3: "Eliminar Reporte por pérdida de libros"			
Flujo básico Eliminar Reporte por pérdida de libros.			
	Actor	Sistema	
1.	Selecciona la opción de eliminar un Reporte por pérdida de libros.		
2.		Muestra un listado con todos los Reporte o por pérdida de libros existentes en la base de datos.	
3.	Selecciona el Reporte por pérdida de libros que desee eliminar.		
4.		El Reporte por pérdida de libros del sistema es eliminado con éxito y se muestra el mensaje : "Reporte por pérdida de libros eliminado satisfactoriamente" termina el caso de uso.	
Sección 4: "Listar Reporte por pérdida de libros"			
Flujo básico Listar Reporte por pérdida de libros			
	Actor	Sistema	
1.	Selecciona la opción listar Reporte por pérdida de libros.		
2.		Muestra un listado con todos los Reporte por pérdida de libros existentes en la base de datos donde se incluye por cada uno su Nombre y apellidos del estudiante, descripción, importe..	
Relaciones			
		CU Incluidos	N/A
		CU Extendidos	N/A
Requisitos no funcionales		Deben predominar los colores blanco y negro.	
Referencias		N/A	
Asuntos pendientes		N/A	
Prototipo de interfaz gráfica de usuario			

Modelo por pérdida de libros

Nombre y apellidos:

Descripción:

Importe:

Fecha:

Aceptar Cancelar

2.3.2 *Diagrama de clases del análisis*

A continuación, se presenta el diagrama de clases del análisis, que se encarga de especificar los elementos de un modelo conceptual temprano. En el presente diagrama intervienen las clases interfaces, las clases controladoras y las clases entidades presentes en el desarrollo de la propuesta, donde se muestra además la interacción entre las clases una vez que el usuario solicite un servicio. Por ejemplo, una vez que el usuario en la clase interfaz *CI_PaginaPrincipal* solicite realizar una acción, y acceda a la clase interfaz correspondiente (por ejemplo, *CI_CrearInventario*), esta clase se comunica con la clase controladora que se encarga de gestionar la solicitud y solicitar a la clase entidad los datos necesarios.

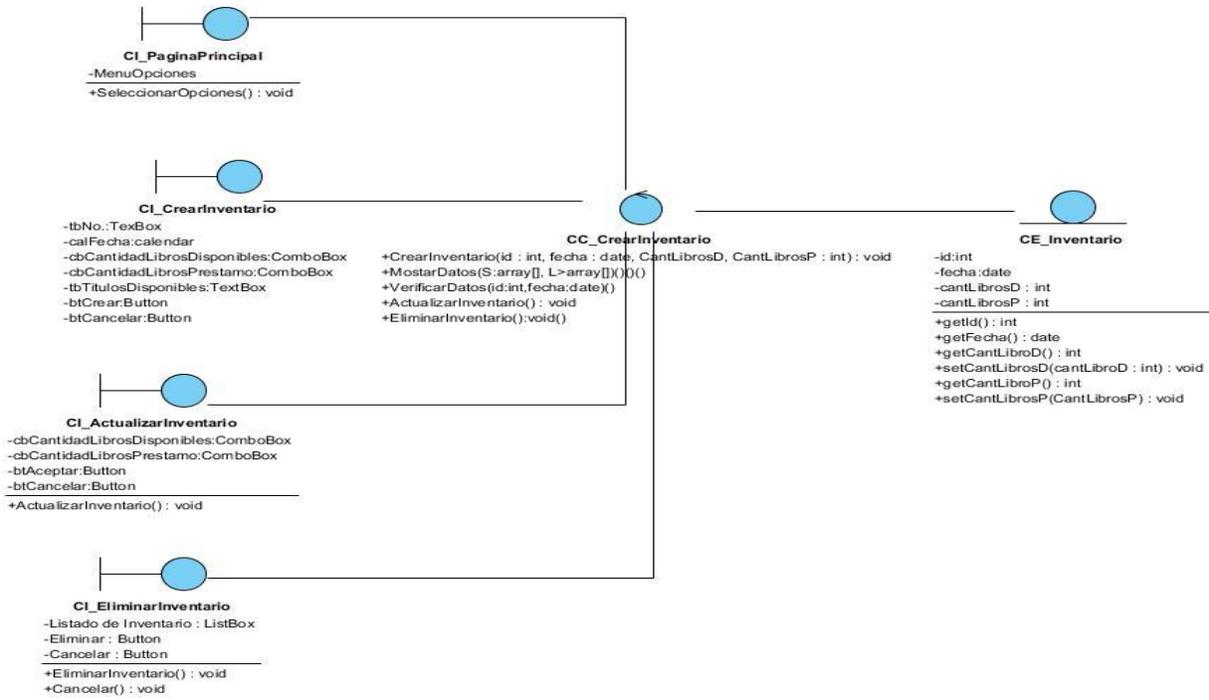


Fig. 7. Diagrama de clases del análisis caso de uso "Gestionar Inventario".

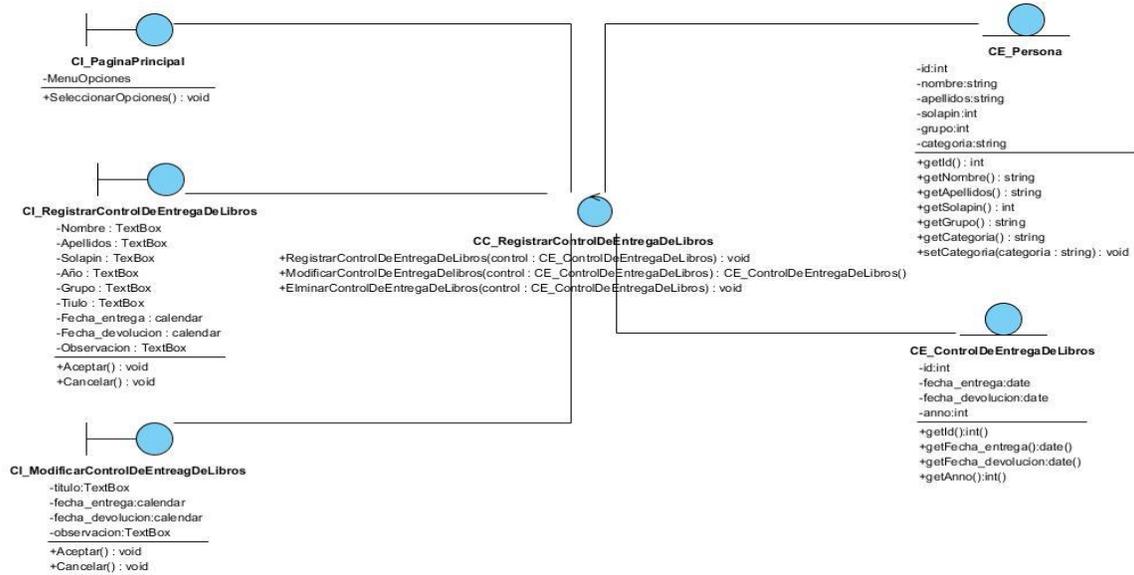


Fig. 8. Diagrama de clases del análisis del caso de uso "Gestionar control de entrega de libros".

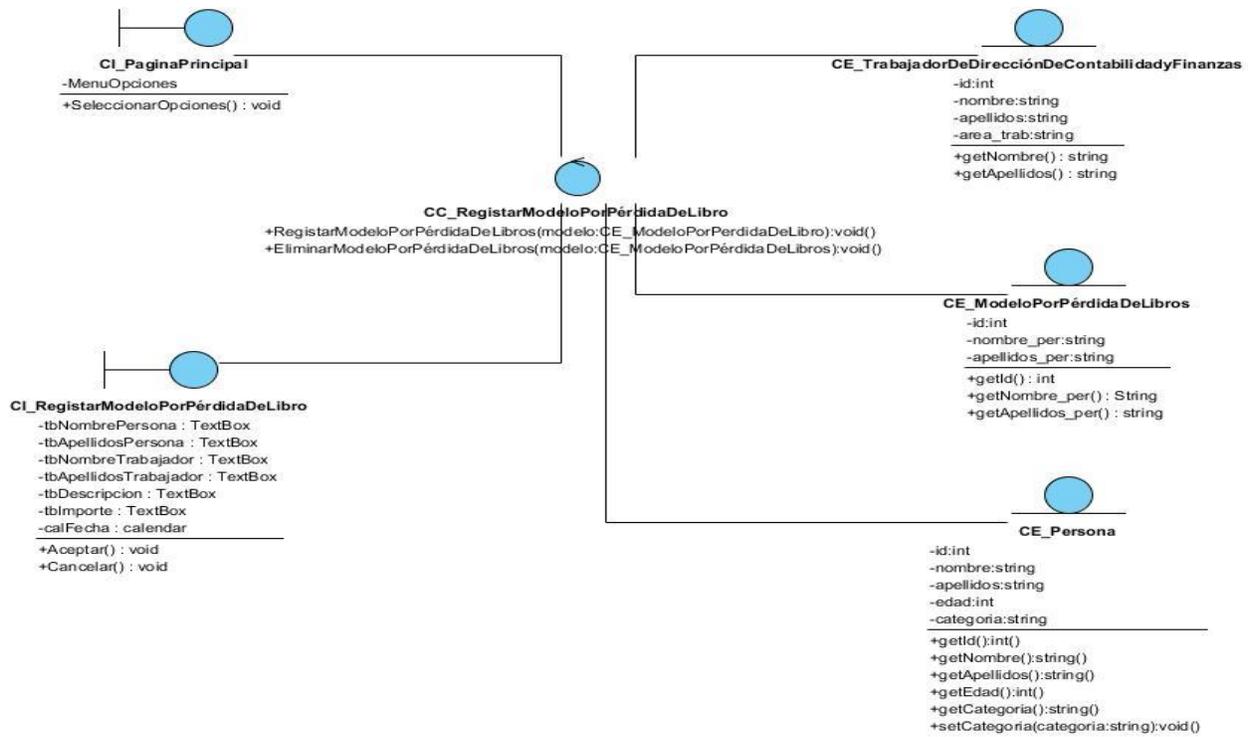


Fig. 9. Diagrama de clases del análisis del caso de uso "Gestionar Reporte por pérdida de libros".

2.3.3 Diagrama de clases del diseño

A continuación, se presentan los diagramas de clases del diseño que muestran las clases que intervienen en la propuesta de solución como sistema y su relación entre ellas. A continuación, se muestra el diseño de la aplicación web utilizando extensiones de UML. Este diagrama está compuesto por la server page (página servidor), (por ejemplo, sp_CrearInventario) que es la página web que contiene el código que se ejecuta en el servidor, que se encarga de construir las client page (páginas clientes), (por ejemplo, cp_CrearInventario, cp_ActualizarInventario y cp_EliminarInventario) las cuales son páginas web con formato HTML interpretadas por el navegador. Cada client page tiene una colección de elementos de entrada que son los formularios, estos formularios se encargan de enviar los valores de sus campos al servidor para ser procesados por la página servidor.

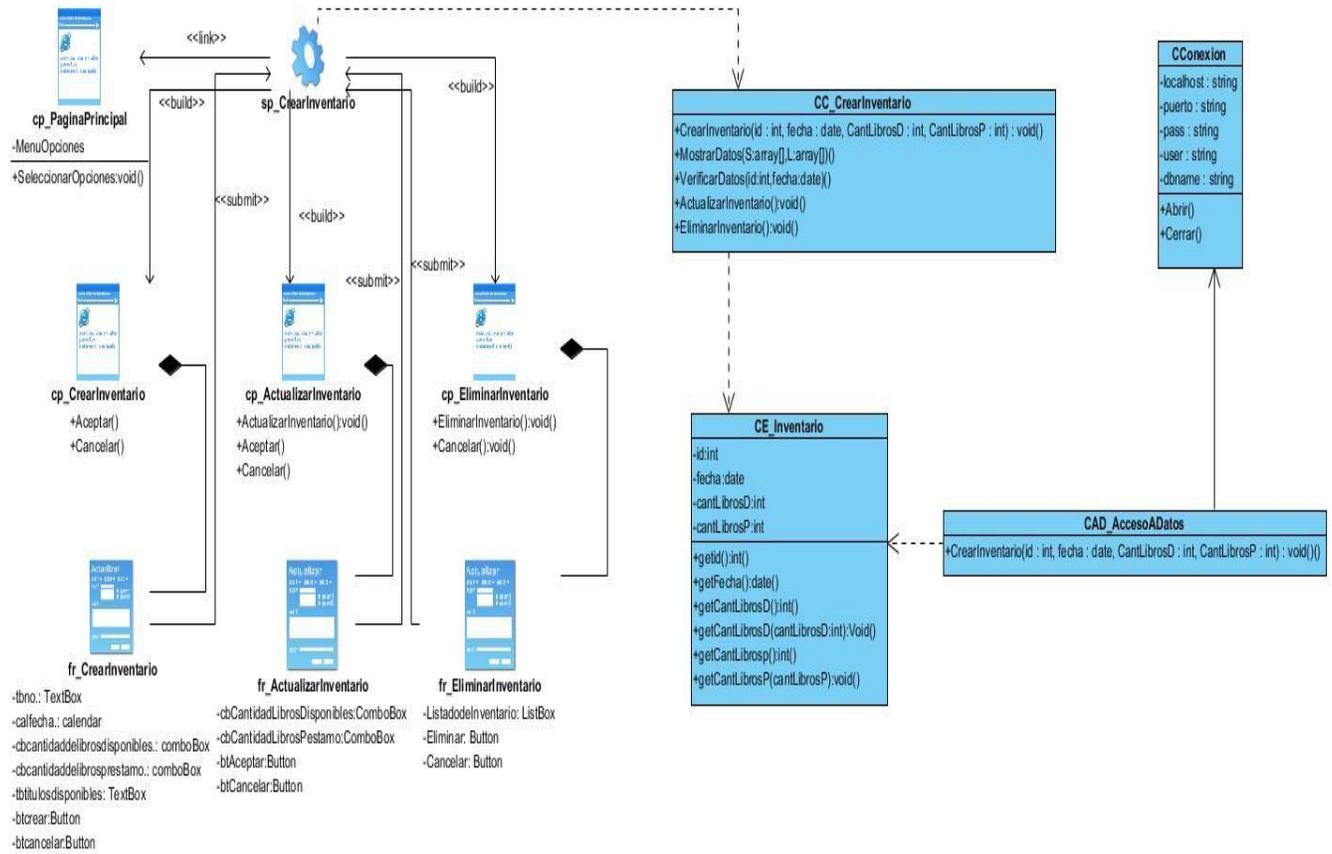


Fig. 10. Diagrama de clases del diseño del caso de uso "Crear Inventario".

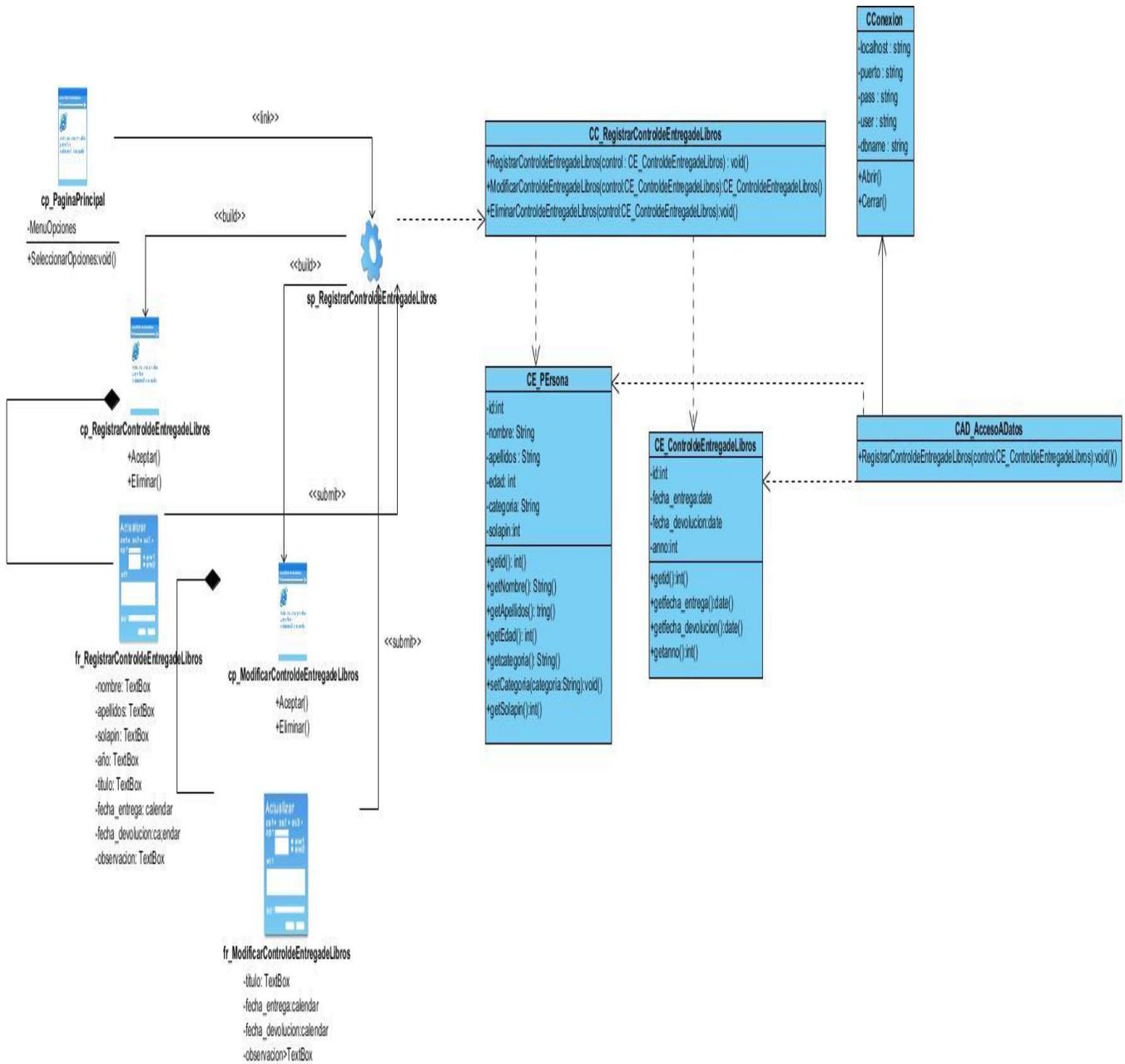


Fig. 11. Diagrama de clases del diseño del caso de uso "Adicionar control de entrega de libros".

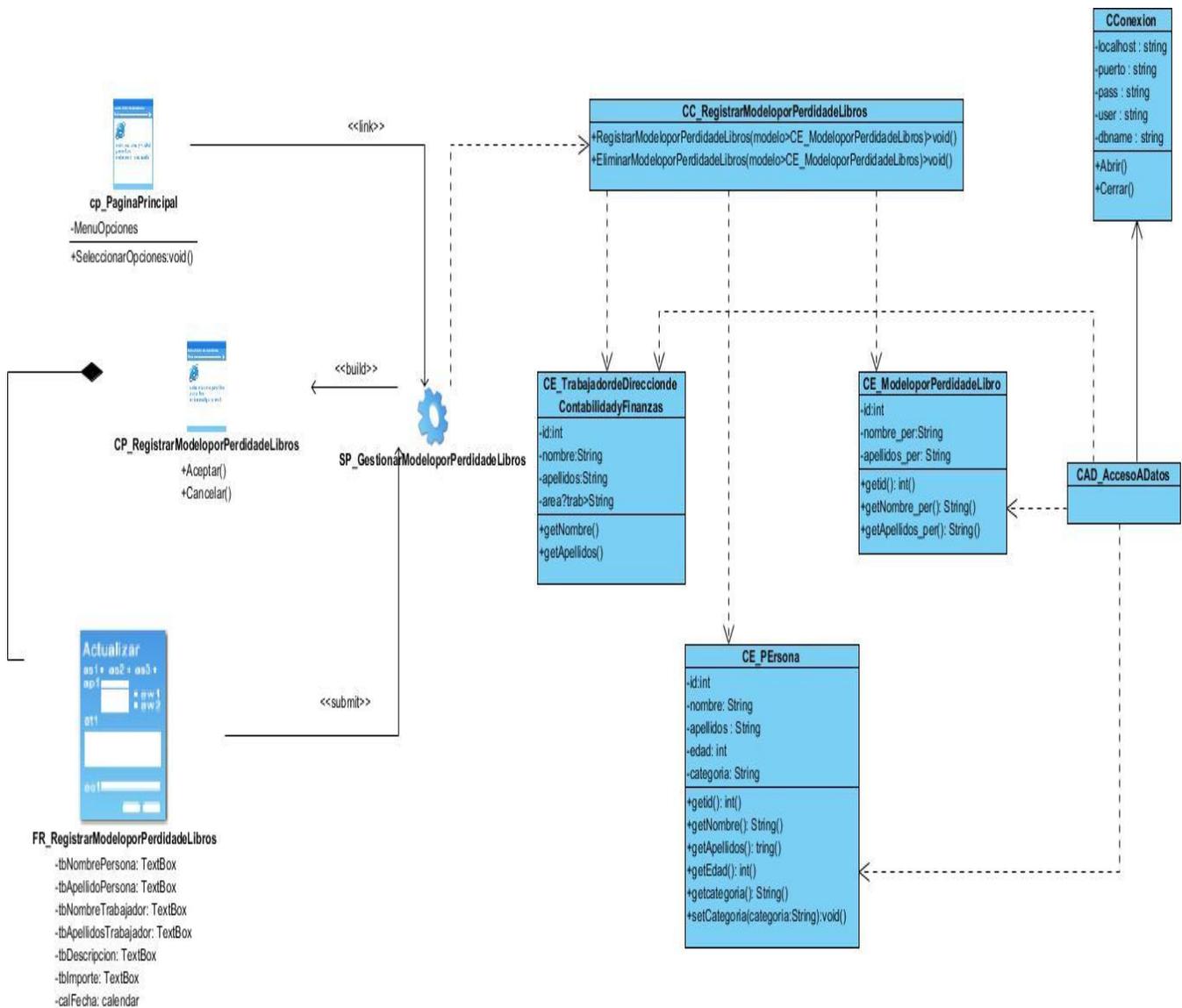


Fig. 12. Diagrama de clases del diseño del caso de uso "Registrar reporte por pérdida de libros".

2.4 Arquitectura de la solución

El Modelo Vista Controlador (MVC) es una arquitectura donde el sistema se estructura en tres componentes lógicos que interactúan entre sí, permitiendo estructurar el código de forma organizada. Se encarga de separar la lógica de negocio de la interfaz de usuario, lo cual facilita la evolución por separado de ambos aspectos e incrementa la reutilización y flexibilidad para el desarrollador. Como parte de la propuesta de solución de esta investigación se estará trabajando con MVC dado que Symfony es un framework diseñado para optimizar el desarrollo de las aplicaciones web basado en el patrón Modelo Vista Controlador (Symfony framework: ¿Cómo desarrollar una aplicación con Symfony?, 2019).

Modelo: El componente modelo maneja los datos del sistema y las operaciones asociadas a esos datos. Se encarga de representar la información con la que trabajará la aplicación, resolviendo la lógica de negocio y el acceso a los datos.

Controlador: El componente controlador (componente del lado del servidor que maneja los requerimientos de HTTP) es el encargado de dirigir la interacción del usuario y pasar estas interacciones a Vista y Modelo. Es responsable de manejar cada solicitud de entrada en la aplicación. El controlador actúa como un enlace entre el modelo y la vista, recibiendo las entradas del usuario y decidiendo cómo actuar dependiendo de las entradas recibidas.

Vista: El componente vista define y gestiona cómo se presentan los datos al usuario. Tanto la vista como el controlador dependen del modelo, el cual no depende de las otras clases. Esta separación permite construir y probar el modelo independientemente de la representación visual (Pressman, 2010).

Symfony como framework, aunque utiliza el patrón de diseño Modelo Vista Controlador, tiene su propia forma de trabajo. En el caso de Symfony 4.0 en línea con MVC, los archivos de la aplicación se distribuyen en carpetas según su función. La estructura que propone Symfony 4 es opcional, si no se requiere una carpeta, no es necesario crear al directorio (Symfony framework:¿Cómo desarrollar una aplicación con Symfony?, 2019).

En el caso de la propuesta de solución que se desea brindar, el usuario se encarga de realizar la solicitud de un servicio en la interfaz principal, la solicitud es gestionada a través del controlador, que realiza la petición de información al modelo, a partir de esta petición el modelo envía los datos al controlador, que se encarga de enviar los datos a la vista para mostrar la información solicitada al usuario. Por ejemplo, una vez que el usuario solicita la creación de un Inventario, esta solicitud es recibida por la CI_PáginaPrincipal en la capa de presentación, la solicitud es gestionada a través de la CC_CrearInventario que se encarga de solicitar la petición de datos a la CE_Inventario en la capa de modelado, datos que son posteriormente enviados a la vista para ser mostrados al usuario. A continuación, en la figura 13 se muestra el funcionamiento de la arquitectura MVC.

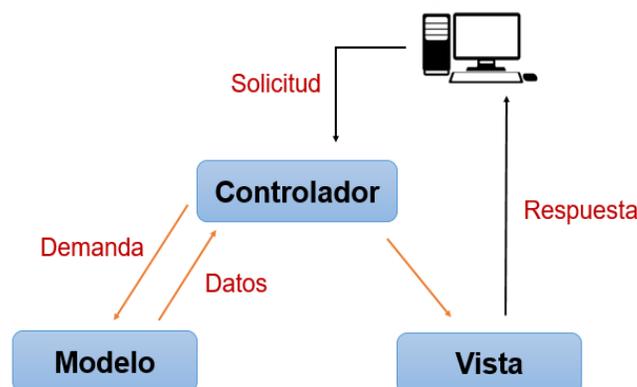


Fig. 13. Representación de la arquitectura MVC.

2.5 Patrones de diseño

Los patrones de diseño son soluciones estandarizadas enfocadas a resolver problemas comunes del desarrollo de software. Los Patrones Generales de Software para Asignación de Responsabilidades más conocidos como patrones GRASP por sus siglas en inglés, se encargan de asignar responsabilidades a las clases y decidir como estas clases deben interactuar. A continuación se relacionan los patrones GRASP seleccionados:

Experto: Se utilizó para asignar labores específicas las clases que contienen la información necesaria para realizarlas. Para la implementación de la propuesta de solución se cuentan con clases controladoras, modelos y entidades, que contienen funcionalidades específicas de acuerdo a los procesos que gestionan, por ejemplo (CC_CrearInventario, CC_RegistrarModeloPorPérdidaDeLibros).

Creador: El patrón creador permite crear las dependencias mínimas necesarias entre las clases favoreciendo el mantenimiento del sistema, por lo que se usó para otorgar responsabilidades a las clases encargadas de crear objetos, por ejemplo (CE_Inventario, CE_ControlEntregaDeLibros CE_ModeloPorPérdidaDeLibros).

Controlador: La propuesta de solución cuenta con clases controladoras, modelos y entidades, por tanto, se utilizó el patrón controlador para asignar la responsabilidad de controlar el flujo de eventos del sistema a clases específicas, obteniendo una separación de la lógica de presentación y la lógica de negocio, por ejemplo (CC_CrearInventario, CC_ControlEntregaDeLibros).

Alta Cohesión: Cada elemento dentro de la propuesta de solución debe realizar una labor única y auto-identificable como parte del sistema. Se utiliza este patrón para agrupar funciones comunes en clases individuales para que un elemento realice operaciones relacionadas entre sí.

Bajo Acoplamiento: Diseñar bajo este patrón permite obtener pocas dependencias entre clases, lo cual tiene como resultado la implementación de un software reutilizable. La utilización de este patrón conlleva a la menor dependencia entre clases lo cual reduce las afectaciones ante posibles modificaciones en alguna de ellas. En la propuesta de solución se evidencia mediante la independencia entre los distintos módulos del sistema.

2.5.1 Diagrama de colaboración

A continuación, se presenta el diagrama de colaboración que muestra interacciones entre objetos en la propuesta de solución como sistema, específicamente muestra las interacciones organizadas alrededor de los roles dentro del sistema. Una vez que el usuario realice una solicitud a la clase interfaz CI_PáginaPrincipal esta envía la petición a la clase controladora, que se encarga de enviar la petición a la

clase interfaz correspondiente, para que sean introducidos los datos por el usuario. Estos datos son enviados a la clase controladora donde son validados en dependencia de la información existente en la clase entidad. Una vez realizada la acción, el usuario va a recibir un mensaje de confirmación en el caso de que la solicitud y los datos introducidos hayan sido procesados correctamente o un mensaje de error en caso de que los datos sean incorrectos.

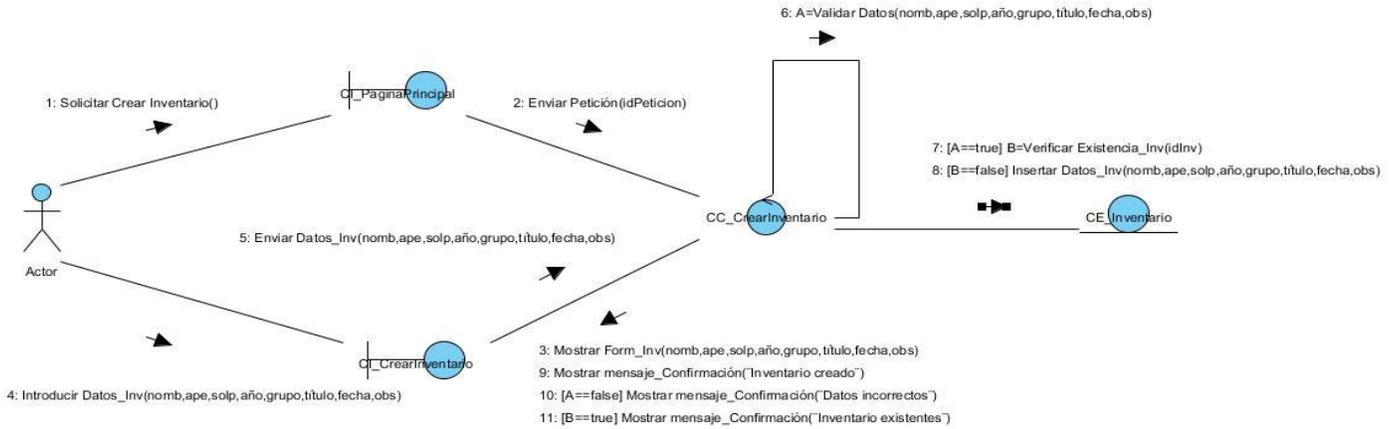


Fig.14. Diagrama de colaboración. Escenario "Crear Inventario".

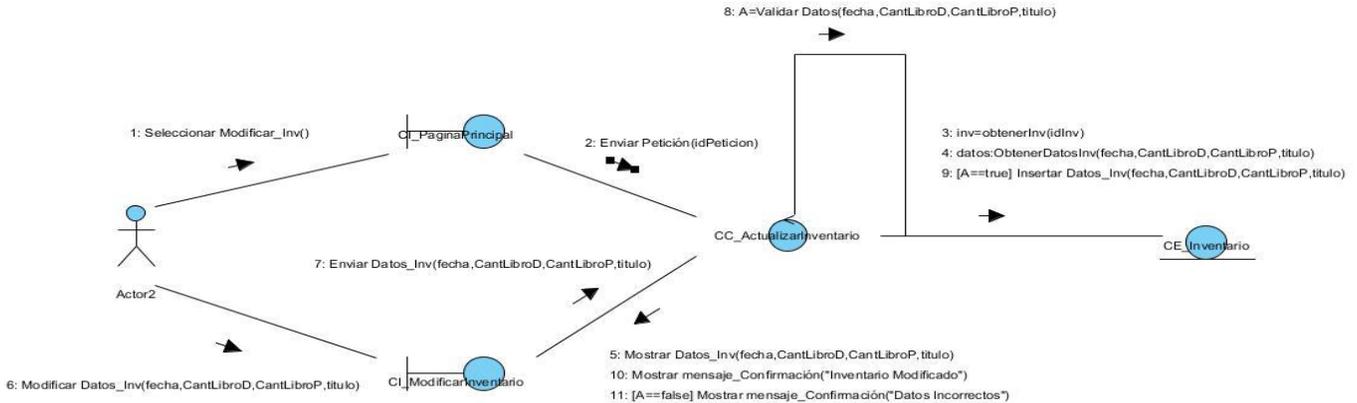


Fig. 15. Diagrama de colaboración. Escenario "Modificar Inventario".

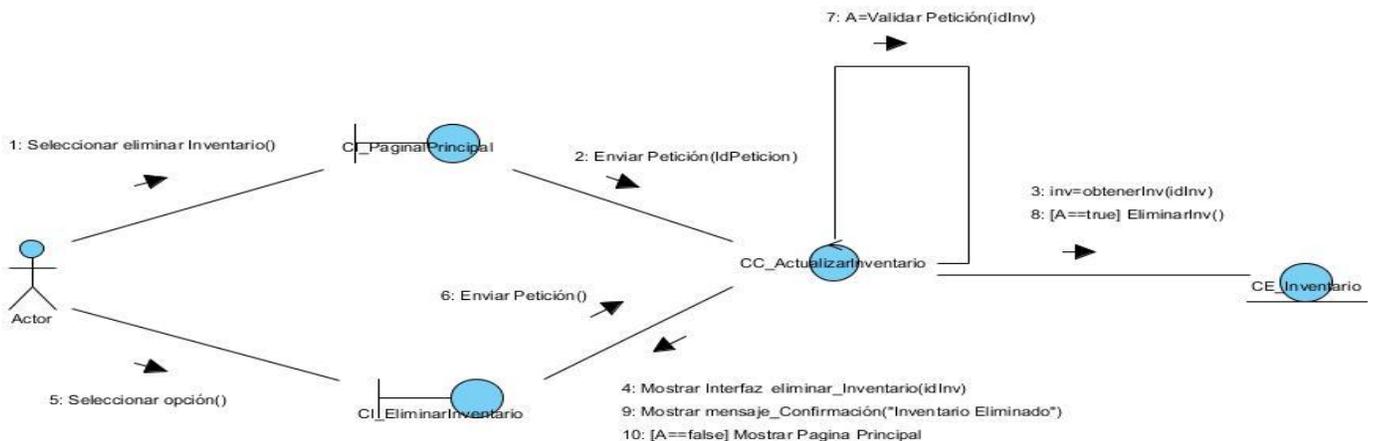


Fig. 16. Diagrama de colaboración. Escenario "Eliminar Inventario".

2.6 Modelo Entidad- Relación

El modelo entidad relación representa la distribución física de la base de datos, donde se reflejan las clases persistentes para la propuesta de solución como sistema. A continuación, se presenta el diagrama de modelo de base de datos de la propuesta.

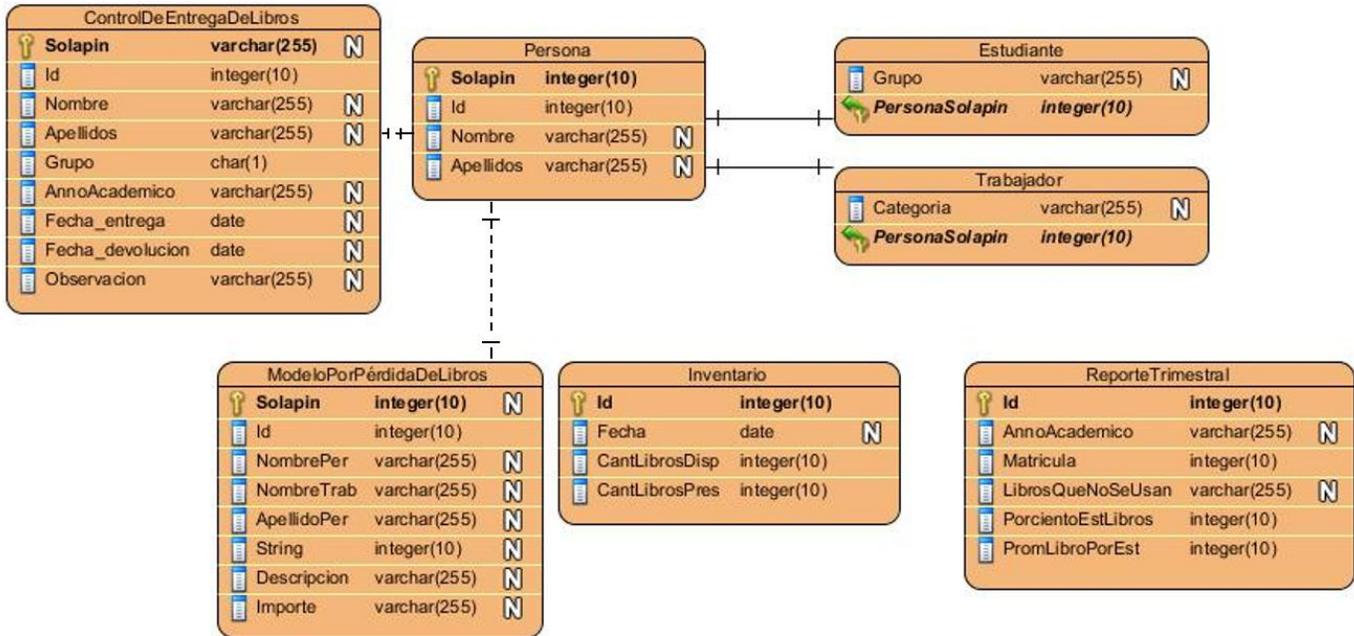


Fig. 17. Diagrama de modelo de base de datos.

2.7 Estructura del sistema

La implementación es la fase donde se materializan los requisitos y se ponen en práctica las descripciones y arquitecturas propuestas en la fase de análisis y diseño. El sistema de gestión como propuesta de solución se encuentra estructurado de acuerdo a la siguiente distribución:

1. **my project:** contiene los archivos de la aplicación divididos según su función en los siguientes subdirectorios: *tests*, *templates*, *config*, *src*, *var*.
2. **tests:** contiene los archivos de prueba de la aplicación.
3. **templates:** contiene las plantillas de la aplicación.
4. **config:** contiene los archivos de configuración, donde se crea un archivo por paquete y por entorno y contiene además un archivo para el enrutamiento.

5. **src**: contiene el núcleo del sistema dividido en los siguientes subdirectorios: *controller*, *entity*, *forms*, *repository*, *migrations*, *validators*.

5.1 **controller**: contiene las clases controladoras del sistema.

5.2 **entity**: contiene las clases entidades del sistema.

5.3 **forms**: contiene los formularios que se utilizan en el sistema.

5.4 **repository**: contiene los archivos encargados de extraer las entidades de las clases.

5.5 **migrations**: contiene los archivos que estructuran la base datos y sus modificaciones.

5.6 **validators**: contiene los archivos encargados de la validación en la aplicación.

6. **var**: incluye los archivos temporales, dentro de los cuales se encuentra la caché.

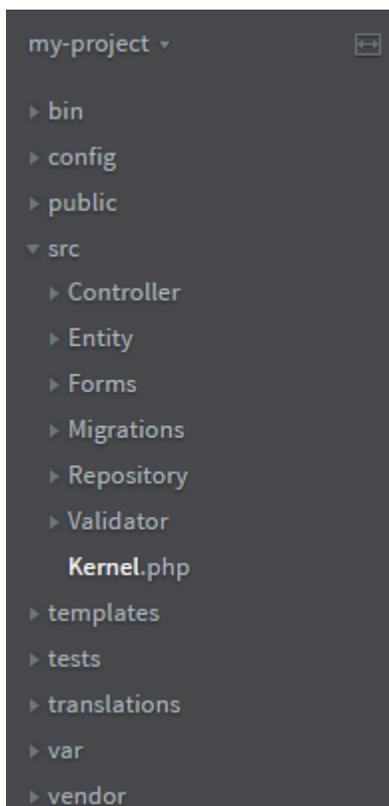


Fig. 18. Directorio del sistema.

2.8 Estándar de codificación de Symfony

El empleo de estándares de codificación posibilita obtener un producto final legible y escalable, respetando la filosofía del framework con el fin de obtener la construcción de un código más universal siguiendo las reglas de codificación de Symfony. PSR significa PHP Standards Recommendation, que son una serie de estándares para garantizar la interoperabilidad entre proyectos. Symfony es un framework totalmente configurable que sigue los estándares definidos en los documentos PSR-0, donde la alternativa es PSR-4

(Lázaro, 2018). Este PSR describe una especificación para clases que serán autocargadas desde directorios de archivos y plantea:

- El término *class* se refiere a clases, interfaces, traits y otras estructuras similares.
- Un nombre *fully qualified class* tiene la siguiente forma: `\(\)*\`
- Las barras bajas no tienen un significado especial en ninguna porción del nombre *fully qualified class*.
- Caracteres alfabéticos en el nombre *fully qualified class* pueden ser cualquier combinación de mayúsculas y minúsculas.
- Todos los nombres de clases deben referenciarse teniendo en cuenta mayúsculas y minúsculas.

Estructura de codificación: El framework Symfony trabaja con un amplio uso de símbolos. El framework hace uso de llaves `{}` para indicar una estructura de control, independientemente del número de declaraciones que contenga. Symfony trabaja con operadores unarios adyacentes a la variable afectada por ejemplo (`!`, `--`, `...`), además de otorgar un espacio después de cada operador binario (`==`, `&&`, `...`) con la excepción del operador de concatenación (`.`). Symfony declara una clase por archivo y define propiedades antes que métodos. El framework además declara métodos `public` primero, luego métodos `protected` y finalmente los métodos `private`. A continuación, se presenta la figura 19 que muestra un fragmento de código de la propuesta de solución.

```
public function new(Request $request)
{
    $task = new Task();
    $task->setTask('Adicionar control de entrega de libros');

    $form = $this->createFormBuilder($task)
        ->add('task', TextType::class)
        ->add('save', SubmitType::class, ['label' => 'Create Task'])
        ->getForm();

    return $this->render('task/new.html.twig', [
        'form' => $form->createView(),
    ]);
}
```

Fig. 19. Fragmento de código 1.

2.9 Diagrama de despliegue

El diagrama de despliegue muestra las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. A continuación, se presenta el modelo de despliegue que muestra el entorno de ejecución física del sistema, donde se muestran los nodos PC_Cliente, Servidor de app (Servidor de la aplicación) y Servidor BD, el cual contiene la información de la base de datos de la propuesta de solución.

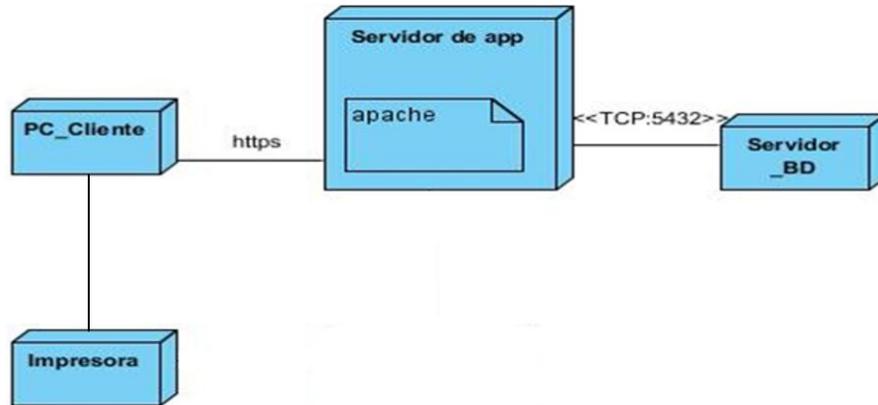


Figura 17. Modelo de despliegue.

2.10 Pruebas de software

Las pruebas de software son un conjunto de actividades que pueden planearse por adelantado y realizarse de manera sistemática. Durante el proceso de software debe definirse una plantilla para la prueba del software: un conjunto de pasos que incluyen métodos de prueba y técnicas de diseño de casos de prueba específicos (Pressman, 2010).

2.11 Verificación y validación de funcionalidades

La creciente preocupación por la calidad en la industria del software tiene como objetivo principal el desarrollo sistemático de productos y servicios de mejor calidad y el cumplimiento de las necesidades y expectativas de los clientes. El propósito de la verificación es asegurar que los productos de trabajo seleccionados cumplen sus requerimientos especificados (Modelo de verificación y validación basado en CMMI, 2013).

El área de proceso de la verificación implica: preparación de la verificación, realización de la verificación e identificación de acciones correctivas. La verificación del producto de trabajo incrementa substancialmente la probabilidad de que el entregable final cumpla con los requerimientos del cliente, del producto y de los componentes del mismo (Modelo de verificación y validación basado en CMMI, 2013).

El modelo de CMMI tiene 22 áreas de procesos en la versión que integra desarrollo de software e ingeniería de sistema (CMMI-SE/SW). Las actividades de validación pueden aplicarse a todos los aspectos del producto en cualquiera de sus entornos previstos (Modelo de verificación y validación basado en CMMI, 2013). El objetivo del proceso de verificación y valoración es establecer la seguridad de que el sistema de software cumple los requisitos establecidos.

2.12 Estrategia de pruebas

Una estrategia de pruebas de software es un conjunto de pasos planificados que integran las técnicas de diseño de casos de prueba que se ejecutan para garantizar la correcta construcción y validación del software.

2.12.1 *Niveles de prueba*

Un nivel de prueba establece las técnicas adecuadas para realizar las pruebas en cada nivel. El empleo de niveles de prueba tiene como beneficio certificar un producto de software para entregar al cliente un producto con el mayor nivel de calidad. Para el desarrollo de esta investigación se establecen los siguientes niveles de prueba.

Pruebas de sistema: Las pruebas de sistemas son una serie de pruebas cuyo propósito principal es ejercitar por completo el sistema basado en computadora. Cada prueba tiene un propósito diferente, para verificar que los elementos se hayan integrado de manera adecuada en todo el sistema funciona y que se realicen las funciones asignadas (Pressman, 2010).

Pruebas unitarias: Las pruebas unitarias se consideran adjuntas al paso de codificación. Las pruebas de unidad se enfocan en la lógica de procesamiento interno y de las estructuras de datos dentro de las fronteras de un componente. Este tipo de prueba puede realizarse en paralelo para múltiples componentes (Pressman, 2010).

CONCLUSIONES

- El análisis de los conceptos analizados en relación a la investigación posibilita una mayor comprensión sobre el campo de acción sobre el cual se desarrolla la presente investigación.
- El estudio de los sistemas homólogos analizados permite determinar las principales fortalezas y debilidades de estos sistemas en comparación con la solución que se desea brindar.
- La definición de la metodología RUP como metodología para guiar el desarrollo de la investigación permite una evaluación de los avances del proyecto por su gestión debido a que basa en el modelo incremental o evolutivo el cual se realiza por pequeños incrementos y en cada uno se puede refinar el anterior.
- Las herramientas y tecnologías usadas permiten establecer las bases tecnológicas para dar solución al problema de investigación.
- La recolección y especificación de requisitos que expresan la necesidad del cliente posibilita identificar la funciones que debe realizar la propuesta de solución como sistema y a su vez determinar qué características deben predominar en el sistema.
- La utilización del Modelo Vista Controlador como arquitectura de solución permite establecer una separación entre la interfaz, la lógica de negocio y los datos y a su vez mantener una estructura predefinida y organizada de la interacción entre estos tres componentes antes mencionados.
- La definición de los patrones de diseño permite establecer la base estructural de la implementación de la propuesta de solución como sistema.
- La definición de los estándares de codificación a utilizar permite lograr un mejor entendimiento en el código del sistema y facilita su futuro mantenimiento.
- El diagrama de despliegue permite una mejor comprensión del entorno de ejecución física de la propuesta de solución.
- El diseño de una estrategia de pruebas permite la detección de errores en el sistema, lo cual permite entregar un producto que cumpla con las necesidades del cliente.

Bibliografía

Fombellida Moreno, Javier y Fernández Muñoz, Javier. 2012. *Sistema de Gestión y control de préstamo de.* Madrid : s.n., 2012.

Adobe. 2019. Brackets. [En línea] 22 de agosto de 2019. [Citado el: 2 de noviembre de 2019.] <https://brackets.uptodown.com/windows>.

Asociación de Academias de la Lengua Española. 2019. Real Academia Española. [En línea] 2019. [Citado el: 13 de Noviembre de 2019.] <https://dle.rae.es/libro>.

Asociación Española de Comprensión Lectora. 2019. Asociación Española de Comprensión Lectora. [En línea] 2019. <https://www.comprensionlectora.es/index.php/2013-11-27-16-50-54/2013-11-27-19-18-32/origen-del-libro>.

Biblioteca del Instituto de Profesores "Artigas" . Biblioteca "Prof. Dr. Antonio M. Grompone". [En línea] <http://ipabiblio.blogspot.com/p/nuevo-reglamento-de-prestamo-1912.html>.

Biblioteca Univeristaria. 2017. Biblioteca Universitaria. Universidad de las Palmas de Gran Canaria. [En línea] 23 de junio de 2017. https://biblioteca.ulpgc.es/conocenos_documentos_reglamentodeprestamo.

Biblioteca Universidad de Sevilla . 2019. PRESTO: Servicio de Préstamo de Libros-e: APP. *Biblioteca Universidad de Sevilla* . [En línea] 15 de noviembre de 2019. <https://guiasbus.us.es/prestamolibro/APP>.

Carmona, Marcel Curbelo. 2013. *Herramienta web para la gestión de la base material de estudio en la Universidad de Pinar del Río "Hermanos Saíz Montes de Oca"*. Pinar del Río : s.n., 2013.

Corbin, Juan Armando. Psicología y Mente. [En línea] <https://psicologiaymente.com/cultura/tipos-de-libros>.

G-On Sistemas Informáticos S.L. Acuña Gestiona tu Centro. [En línea] <http://www.gestionatucentro.com/>. *Informatización del servicio de préstamo de literatura docente. Amador, Pedro Leonardo Pérez, y otros. 2017.* no.5, Pinar del Río : Princesa, 2017, Vol. vol.21.

Instituto Superior de Formación y Recursos en Red para el Profesorado. 2008. P'aginas web. [En línea] 2008. [Citado el: 24 de agosto de 2020.] <https://www.uv.es/fragar/html/html0103.html>.

Jaramillo, Wendy. 2016. *Aplicación de la Metodología RUP y el patrón de diseño MVC en la construcción de un sistema de gestión académica para la Unidad Educativa Ángel de la Guarda*. Quito : s.n., 2016.

Lázaro, Diego. 2018. Libro de PHP. [En línea] 2018. [Citado el: 17 de agosto de 2020.] <https://diego.com.es/interoperabilidad-y-psrs-en-symfony>.

León, Álvaro de. 2019. infranetworking. [En línea] 22 de junio de 2019. [Citado el: 22 de abril de 2020.] <https://blog.infranetworking.com/que-es-apache-servidor/>.

LinkedIn Learning. 2012. SlideShare. [En línea] 12 de febrero de 2012. <https://es.slideshare.net/LeerEsDivertido/los-librossus-partes-y-tipos>.

López, Brayan Salazar. 2016. INGENIERIAINDUSTRIALONLINE.COM. [En línea] 2016. [Citado el: 22 de Noviembre de 2019.] <https://www.ingenieriaindustrialonline.com/herramientas-para-el-ingeniero-industrial/gesti%C3%B3n-de-almacenes/>.

Modelo de verificación y validación basado en CMMI. **Puello, Osvaldo. 2013.** No. 1, Colombia : s.n., 2013, Vol. Vol. 1.

Morell, Melchor Gil. 2008. *Resolución No. 858/ 08 Procedimiento para el Control de los Medios de la Universidad.* La Habana : s.n., 2008.

Noega Systems.Soluciones de almacenaje. 2007. NOEGA Systems - Sistemas de Almacenaje y Logística Interna. *NOEGA Systems - Sistemas de Almacenaje y Logística Interna.* [En línea] 1 de febrero de 2007. [Citado el: 10 de enero de 2020.] <https://www.noegasystems.com/blog/logistica/almacen-funciones-actividades-planificacion-ubicacion>.

php.net. 2017. php. *Migrar de PHP 5.6.x a PHP 7.0.x.* [En línea] 2017. [Citado el: 5 de enero de 2020.] <https://www.php.net/manual/es/migration70.php>.

Pressman, Roger S. 2010. *Ingeniería del Software. Un enfoque práctico. 7ma Edición.* México : The McGraw-Hill, 2010. 978-607-15-5473-3.

Real Academia Española. 2019. Real Academia Española. *Real Academia Española.* [En línea] 2019. [Citado el: 10 de enero de 2020.] <https://dle.rae.es/almac%C3%A9n>.

Sanz, Roberto. 2017. Nextop. *Nextop.* [En línea] 22 de agosto de 2017. [Citado el: 12 de abril de 2020.] https://nextop.es/principios-basicos-de-scrum-metodologias-agiles/#Priorizar_las_tareas_del_Proyecto_que impliquen_mayor_valor_de_negocio.

SensioLabs. 2017. Symfony. [En línea] 23 de mayo de 2017. [Citado el: 27 de octubre de 2019.] <https://symfony.com/>.

Symfony framework:¿Cómo desarrollar una aplicación con Symfony? **Acosta, Víctor Manuel. 2019.** Granada : INESEM, 2019.

The PostgreSQL Global Development Group. 2010. PostgreSQL: The World's Most Advanced Open Source Relational Database. [En línea] 20 de septiembre de 2010. [Citado el: 2 de noviembre de 2019.] <https://www.postgresql.org/about/news/1235/>.

Una mirada al mundo de las bibliotecas. 2012. Una mirada al mundo de las bibliotecas.Web de difusión cultural. *Una mirada al mundo de las bibliotecas.Web de difusión cultural.* [En línea] 23 de 6 de 2012.

[Citado el: 10 de enero de 2020.] <https://www.alquiblaweb.com/2012/06/23/las-bibliotecas-en-la-antiguedad-5/>.

Visual Paradigm. 2010. Visual Paradigm . [En línea] 16 de agosto de 2010. [Citado el: 10 de noviembre de 2019.] <https://www.visual-paradigm.com/tw/aboutus/newsreleases/vpuml80.jsp>.