

**ANÁLISIS, MANTENIMIENTO, DISEÑO, DESARROLLO E IMPLEMENTACIÓN  
DE NUEVAS FUNCIONALIDADES Y MEJORAS RELACIONADAS CON LOS  
MÓDULOS DE AULA VIRTUAL Y DE TRABAJOS DE GRADO EN LA  
PLATAFORMA COMA COMUNIDAD ACADÉMICA.**

**KAREN XIMENA OROZCO GAMBOA**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍA FISICOMECAÑICAS  
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA  
BUCARAMANGA  
2018**

**ANÁLISIS, MANTENIMIENTO, DISEÑO, DESARROLLO E IMPLEMENTACIÓN  
DE NUEVAS FUNCIONALIDADES Y MEJORAS RELACIONADAS CON LOS  
MÓDULOS DE AULA VIRTUAL Y DE TRABAJOS DE GRADO EN LA  
PLATAFORMA COMA COMUNIDAD ACADÉMICA.**

**KAREN XIMENA OROZCO GAMBOA**

**Trabajo de grado para optar al título de Ingeniera de Sistemas**

**Director  
Msc. LUIS IGNACIO GONZÁLEZ RAMÍREZ  
Magíster en Informática**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍA FISICOMECÁNICAS  
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA  
BUCARAMANGA  
2018**

## **AGRADECIMIENTOS**

A Dios por acompañarme en cada paso que doy, por su amor incondicional y por darme enseñarme cada día el mejor camino a seguir.

A mi madre, Nelly Gamboa Espejo, por su enorme sacrificio de sacarme adelante a pesar de los obstáculos y apoyarme a cumplir este objetivo de ser profesional.

A mi hermana, Beatriz Orozco Gamboa, por acompañarme, cuidarme y ayudarme a crecer como persona durante toda mi vida.

A mi padre, José de Jesús Orozco Rengifo, que a pesar de que ya no estás conmigo físicamente, se quede dónde te encuentres me apoyas y estas muy orgulloso, me enseñaste muchas lecciones de vida y me apoyaste en cada una de mis decisiones.

A Yessenia Rincón, por estar conmigo y comprenderme en cada momento alegre y difícil, por su apoyo incondicional.

Al profesor Luis Ignacio González, que no solo fue mi director de proyecto, sino también un amigo que compartió todos sus conocimientos y creyó en mi para desarrollar las diferentes actividades del grupo.

A Carlos Díaz, Tania Vergara, Felipe Vesga, Harold Ardila y Fabio Montañez, por brindarme su valiosa amistad desde el primer momento y apoyarme en cada una de mis decisiones para así cumplir este objetivo de ser Ingeniera de Sistemas UIS.

## CONTENIDO

INTRODUCCIÓN .....	13
1. PLANTEAMIENTO DEL PROYECTO .....	14
1.1 DEFINICIÓN DE LA SITUACIÓN ACTUAL.....	14
2. JUSTIFICACIÓN.....	15
2.1 OBJETIVOS.....	16
1.4 IMPACTO Y VIABILIDAD.....	18
3. MARCO TEÓRICO .....	19
3.1 ARQUITECTURA CLIENTE/SERVIDOR.....	19
3.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS .....	23
3.3 BASES DE DATOS.....	26
3.4 NETBEANS.....	28
3.5 SISTEMA DE CONTROL DE VERSIONES .....	29
3.6 PROGRAMACIÓN UTILIZADA.....	30
3.7 SERVIDORES WEB .....	32
4. MARCO METODOLÓGICO .....	33
4.1 PROTOTIPO EVOLUTIVO .....	33
4.2 LENGUAJE DE MODELADO UNIFICADO.....	35
4.3 ESTÁNDARES DE PROGRAMACIÓN.....	39
5. DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO.....	41
5.1 PROTOTIPO ESPERADO.....	41
5.2 DIAGRAMAS DE CASOS DE USO .....	46
5.3 DOCUMENTACION DE CASOS DE USO DEL SISTEMA .....	51
5.4 DISEÑO Y ANÁLISIS.....	58
5.5 DESCRIPCIÓN DE LAS ENTIDADES.....	59
5.6 MODELO DE PROCESOS DEL SISTEMA.....	59
5.7 IMPLEMENTACIÓN, IMPLANTACIÓN Y PRUEBAS GENERALES.....	59

5.8 MANTENIMIENTO Y ADMINISTRACIÓN.....	60
6. PRUEBAS DEL SISTEMA .....	62
6.1 PRUEBAS DE VERIFICACIÓN .....	62
7. CONCLUSIONES .....	69
8. RECOMENDACIONES.....	70
BIBLIOGRAFÍA.....	71
ANEXOS.....	73

## LISTA DE FIGURAS

Figura 1 Modelo Cliente/servidor .....	19
Figura 2 Modelo de acceso a JSP .....	25
Figura 3 Prototipo evolutivo .....	33
Figura 4 Diagramas de Casos de Uso .....	36
Figura 5 Diagrama de Secuencias.....	38
Figura 6 Diagrama de casos de uso: Cancelar proyecto de grado en desarrollo. .	46
Figura 7 Diagrama de casos de uso: Implementar látex al banco de preguntas del aula virtual.....	46
Figura 8 Diagrama de casos de uso: Visualización de preguntas con látex por el estudiante en el aula virtual. ....	47
Figura 9 Diagrama de casos de uso: Visualización de preguntas con látex por el estudiante en el aula virtual. ....	47
Figura 10 Diagrama de casos de uso: Exportar notas en formato csv del aula virtual .....	48
Figura 11 Diagrama de casos de uso: Creación y modificación del libro virtual ....	49
Figura 12 Diagrama de casos de uso: Creación y modificación del libro virtual por un grupo colaborador.....	50
Figura 13 Diagrama E/R: Construcción y modificación del libro virtual.....	58

## LISTA DE TABLAS

Tabla 1 Casos de uso: Cancelar proyecto de grado en desarrollo. ....	51
Tabla 2 Casos de uso: Crear pregunta por un profesor al banco de preguntas del aula virtual.....	52
Tabla 3 Casos de uso: Proponer pregunta por un usuario al banco de preguntas del aula virtual.....	52
Tabla 4 Casos de uso: Visualización de preguntas del taller con látex por el estudiante en el aula virtual. ....	53
Tabla 5 Casos de uso: Visualización de preguntas del examen con látex por el estudiante en el aula virtual. ....	54
Tabla 6 Casos de uso: Exportar notas en formato csv del aula virtual .....	54
Tabla 7 Casos de uso: Creación y modificación del libro virtual por parte del profesor.....	55
Tabla 8 Casos de uso: Creación y modificación del libro virtual por un grupo colaborador. ....	56
Tabla 9 Descripción de las entidades .....	59
Tabla 10 Pruebas Realizadas: Cancelar proyecto de grado en desarrollo. ....	62
Tabla 11 Pruebas Realizadas: Creación de pregunta por un profesor .....	63
Tabla 12 Pruebas Realizadas: Creación de pregunta por un usuario.....	63
Tabla 13 Pruebas Realizadas: Visualización de látex en el taller y examen.....	64
Tabla 14 Pruebas Realizadas: Exportar notas en formato csv del aula virtual .....	64
Tabla 15 Pruebas Realizadas: Creación y modificación del libro virtual .....	65
Tabla 16 Pruebas Realizadas: Creación y modificación del libro por un encargado .....	67

## LISTA DE ANEXOS

Anexo A. Modelo de Procesos del Sistema .....	73
---	----

## RESUMEN

**TÍTULO:** Análisis, mantenimiento, diseño, desarrollo e implementación de nuevas funcionalidades y mejoras relacionadas con los módulos de Aula Virtual y de Trabajos de Grado en la plataforma COMA Comunidad Académica.\*

**AUTOR:** Karen Ximena Orozco Gamboa\*\*

**PALABRAS CLAVE:** Calumet, Portal, Aula, Desarrollo.

### **DESCRIPCIÓN:**

En la actualidad las escuelas cuentan con un sistema de información orientado a la Web que se encarga de la administración y control de las diferentes actividades tanto académicas, como administrativas que se realizan dentro de las escuelas y las Facultades, así como del control de usuarios y servicios que se les proporcionan.

Actualmente el portal cuenta con el servicio de Aula Virtual. Este es un medio donde el docente complementa sus clases. Por lo tanto, se hace necesario en el módulo de notas implementar un servicio que permita generar y descargar en archivos csv las notas de cada tipo de actividad y definitiva, con el fin de hacer más sencilla la labor de subida de notas al sistema de la universidad. Además, el aula virtual se hace necesario agregar en los talleres, exámenes y foros un control de tiempo, con el objetivo de avisar a los usuarios el tiempo restante para que se cierre la sesión.

El servicio del libro virtual, que se construye por parte del docente, se realizaron cambios en las interfaces para hacerlas más amigables e intuitivas a los usuarios.

En el banco de preguntas del aula, se implementa Látex, en los enunciados, proceso y soluciones de cada pregunta, con el fin de soportar diferentes estructuras y notaciones matemáticas.

Debido a solicitud de varias escuelas, en el módulo de trabajos de grado, se crea un nuevo servicio que permite al comité de trabajos de grado, cancelar cualquier proyecto en desarrollo, abriendo la posibilidad de un comentario para justificar la decisión. Además, en este módulo, a solicitud de varias escuelas, se implementó ocultar a los autores el nombre del evaluador de plan asignado.

---

\* Trabajo de grado

\*\* Facultad de Ingenierías Físico-mecánicas. Escuela de Ingeniería de Sistemas e Informática. Director: Msc. Luis Ignacio González Ramírez

## ABSTRACT

**TITLE:** Analysis, maintenance, design, development and implementation of new functionalities and improvements related to the Virtual Classroom and Degree Work modules in the COMA academic community.\*

**AUTHOR:** Karen Ximena Orozco Gamboa.\*\*

**KEYWORDS:** Calumet, Web Portal, Classroom, Development.

### DESCRIPTION

Now schools have an information system oriented to the Web that is responsible for the Administration and control of the various activities both academic, as administrative that are carried out within the schools and faculties, as well as the control of users and services that are provided to them.

Currently the portal has the Virtual Classroom service. This is a medium where the teacher complements his classes. Therefore, it is necessary in the notes module to implement a service that allows to generate and download in csv files the notes of each type of activity and final, in order to make easier the task of uploading notes to the system of the college. In addition, the virtual classroom is necessary to add a time control in the workshops, exams and forums, in order to notify users of the time remaining for the session to close.

The service of the virtual book, which is built by the teacher, changes were made in the interfaces to make them more user friendly and intuitive.

In the question pool of the classroom, Latex is implemented, in the statements, process and solutions of each question, in order to support different structures and mathematical notations.

Due to the request of several schools, in the module of degree works, a new service is created that allows the committee of works of degree, canceling any project in development, opening the possibility of a comment to justify the decision. In addition, in this module, at the request of several schools, the possibility of hiding the name of the assigned plan evaluator from the authors was implemented.

---

\* Degree Work

\*\* Faculty of Physico-Mechanical Engineering. Department of Systems Engineering and Computing Science. Supervisor Msc. Luis Ignacio González Ramírez

## INTRODUCCIÓN

Los Portales Web Comunidad Académica son los principales canales de comunicación e integración entre los miembros de la comunidad de las escuelas y miembros en general de la comunidad UIS. Actualmente los servicios se han extendido a más escuelas de las distintas facultades gracias a la aceptación de los usuarios por lo que se adopta el objetivo de mantener y mejorar los servicios que se ofrecen, además de crear nuevos servicios que satisfagan las necesidades crecientes de los usuarios de la comunidad académica.

El grupo Calumet, grupo de desarrollo de software de la escuela de ingeniería de sistemas, se encarga de diseñar, desarrollar e implementar aplicaciones web y soluciones informáticas con el fin de ayudar y facilitar a las escuelas de la universidad a administrar su información, buscando unir a la comunidad universitaria y brindar seguridad en los procesos e integridad a las personas. Con el fin de cumplir el objetivo principal del grupo y proporcionar portales web con contenido dinámico, se cuenta con herramientas software de libre distribución como Netbeans, Github, JavaScript, JQuery, GNU Linux, entre otros.

En este documento se presenta un soporte teórico, metodológico y técnico del desarrollo realizado en los servicios de: Aula Virtual para facilitar la subida de las notas al sistema de la universidad, mediante archivos csv que se generan en el aula. En el banco del aula se implementa Látex que permita la notación matemática en los enunciados de las preguntas. Se mejora la interfaz de la construcción por parte del docente, del libro virtual, para que sea más amigable a los usuarios. Se agrega un control de tiempo en los talleres, exámenes y foros con el fin de informarle al usuario el tiempo restante para que se cierre la sesión. En el servicio de Trabajo de Grado, se implementa un nuevo servicio que permita al comité de trabajo de grado cancelar cualquier proyecto en desarrollo. Finalmente, en este servicio, se implementa ocultar a los autores el evaluador al plan asignado.

## **1. PLANTEAMIENTO DEL PROYECTO**

### **1.1 DEFINICIÓN DE LA SITUACIÓN ACTUAL**

Los servicios de los portales Web de las escuelas deben mejorar constantemente y adaptarse a los cambios que se presenten en su entorno, a su vez deben dar solución a los problemas y necesidades que surjan por parte de los usuarios del sistema para incrementar su tiempo de vida útil y no llegar a convertirse en un software obsoleto, razón por la cual las labores de mantenimiento y actualización se hacen indispensables.

El aula virtual y los servicios que esta ofrece deben ser actualizados para soportar nuevas necesidades de la comunidad, la actualización consta de cambio de interfaz e implementación de nuevas funcionalidades que permitan a los usuarios interactuar con los servicios Web, y entre ellos mismos, de una mejor manera.

Además, se agrega un nuevo servicio en el módulo de trabajos de grado en el cual puedan cancelar cualquier proyecto en desarrollo abriendo al comité la posibilidad de un comentario para justificar la decisión y ocultar el evaluador de plan asignado a los autores, entre otros cambios funcionales que mejore el desempeño y la atención al usuario.

## 2 JUSTIFICACIÓN

El Portal Web Comunidad Académica (COMA), que desarrolla y mantiene el grupo Calumet, es una plataforma que provee lugares, adicionales a los ya existentes, de encuentro y comunicación a la comunidad de los programas de las diferentes escuelas. Esta plataforma contribuye a la realización de la misión en sus tres ejes: extensión, investigación y docencia. En ella, la Comunidad podrá mantener actualizada su hoja de vida, publicar noticias, proponer eventos, consultar y crear agendas, compartir documentos, enviar correos, y otros servicios que podrán hacer más fácil la realización de la misión institucional. Los servicios de los portales web deben actualizarse constantemente, mejorar y adaptarse a los cambios que se presenten en su entorno, satisfaciendo las necesidades de los usuarios del sistema. Esto permite aumentar el tiempo de vida útil del software evitando que pase a ser obsoleto con la creación de nuevos servicios, y con el mantenimiento realizado a los existentes.

En el módulo de trabajos de grado se hace necesario permitir al Comité de Trabajos de Grado cancelar cualquier proyecto en desarrollo, abriendo al Comité la posibilidad de un comentario para justificar la decisión. Además, en este módulo se ha solicitado por varias escuelas que se dé la posibilidad de ocultar a los autores el nombre del evaluador de plan asignado. Por lo anterior, se debe definir un mecanismo para que el Administrador de Trabajos de Grado pueda para cada programa definir lo anterior.

En el Aula Virtual se realizarán varios cambios:

- Implementar  $\text{\LaTeX}$  en el enunciado de las preguntas del banco del aula para soportar las diferentes estructuras y notaciones matemáticas en el enunciado. Se requiere poder soportar matrices y notaciones de integración y derivación, así como otras, soportadas por  $\text{\LaTeX}$ .

- Agregar el control de tiempo de sesión en los talleres, exámenes y foros con el objetivo de avisar a los usuarios el tiempo restante para que se cierre la sesión. El estudiante podrá decidir si continua o termina la sesión.
- En el módulo de notas permitir a los profesores generar archivos csv para cada tipo de actividad y para la definitiva, con el fin de hacer más sencilla la labor de subida de notas al sistema de la universidad. Actualmente, el profesor debe de manera manual digitar cada una de las notas, labor tediosa y propensa a errores, por lo que este nuevo servicio todo el proceso será mas sencillo.
- Modificar el servicio que construye, por parte del profesor, y muestra los contenidos objeto de estudio, a los estudiantes, para generar interfaces más amigables e intuitivas a los usuarios. En este momento, los estudiantes deben seguir un proceso un tanto largo para acceder a estos recursos. Se propone que para cada tema se listen los recursos con alguna información adicional, y se pueda abrir el archivo en otra pestaña del navegador. A su vez, la interface para crear estos recursos (libro del aula) se debe actualizar para que sea una labor más sencilla. Se debe tener en cuenta, que algunos recursos ya no se soportan por los navegadores como son applets de java. Por lo anterior, se debe de eliminar ese tipo de recurso y crear la posibilidad de una alternativa para construir simuladores, que eran posible con las applets de java.

## 2.1 OBJETIVOS

**2.1.1 Objetivo general.** Realizar las funciones de mantenimiento, análisis, diseño, desarrollo e implementación de nuevas funcionalidades y servicios para el Aula Virtual y para el módulo de Trabajos de Grado con el fin de hacerlos óptimos, eficientes y sostenibles, permitiendo acceder a la información de manera más sencilla, ágil, óptima y eficiente.

## 2.1.2 OBJETIVOS ESPECÍFICOS

- En el módulo de trabajos y tesis de grados permitir al Comité de Trabajos de Grado cancelar cualquier proyecto. Se debe permitir al Comité ingresar comentarios que justifiquen la decisión tomada.
- En el módulo de trabajos de grado, y a solicitud de las Escuelas (Comité de trabajo de grado y Claustro de profesores), permitir a las Escuelas que se oculte a los autores el nombre del profesor evaluador de plan. Esta corrección debe contemplar los servicios de correos, buscadores, actas y mi proyecto.
- Para el banco de preguntas implementar el sistema de composición de texto Látex que permite generar estructuras matemáticas en los enunciados de las preguntas.
- Implementar un sistema de control de sesión en el aula virtual que permita controlar el tiempo y avise al usuario en un momento dado del tiempo restante de sesión para que defina si desea continuar la sesión o cerrarla. Este sistema deberá estar funcional en los talleres, exámenes y foros.
- Generar en formato csv un archivo con las notas de las diferentes actividades que se realizan en el aula con el fin de facilitar la exportación de notas del aula virtual al sistema de la UIS.
- Realizar la construcción y modificación de la forma en que se muestra los servicios de recursos multimedia empleados en el aula virtual, permitiendo menús a los profesores o encargados, para así mostrarle a los estudiantes, los contenidos objeto de estudio que acompañan las diferentes temáticas del aula.

## **2.2 IMPACTO Y VIABILIDAD**

**2.2.1 Impacto.** Los portales web de las escuelas han sido una herramienta útil para el manejo de la información, por lo tanto, es necesario realizar labores de administración y mantenimiento, para ofrecer al usuario un sitio más confiable.

Se pretende que los procesos que se realizan en las escuelas cada día sean más ágiles, dinámicos, seguros y eficientes, permitiendo una mejor organización de la información, razón por la cual se crean nuevos servicios que se ajusten a las necesidades que puedan surgir.

**2.2.2 Viabilidad.** La administración del sitio, soporte a usuarios, mantenimiento y desarrollo de nuevos servicios es viable pues se usará software de libre distribución, recurso humano preparado para tal fin, servidores que marchan de manera legal en las escuelas, equipos disponibles y todo el soporte tecnológico necesario para el desarrollo del mismo.

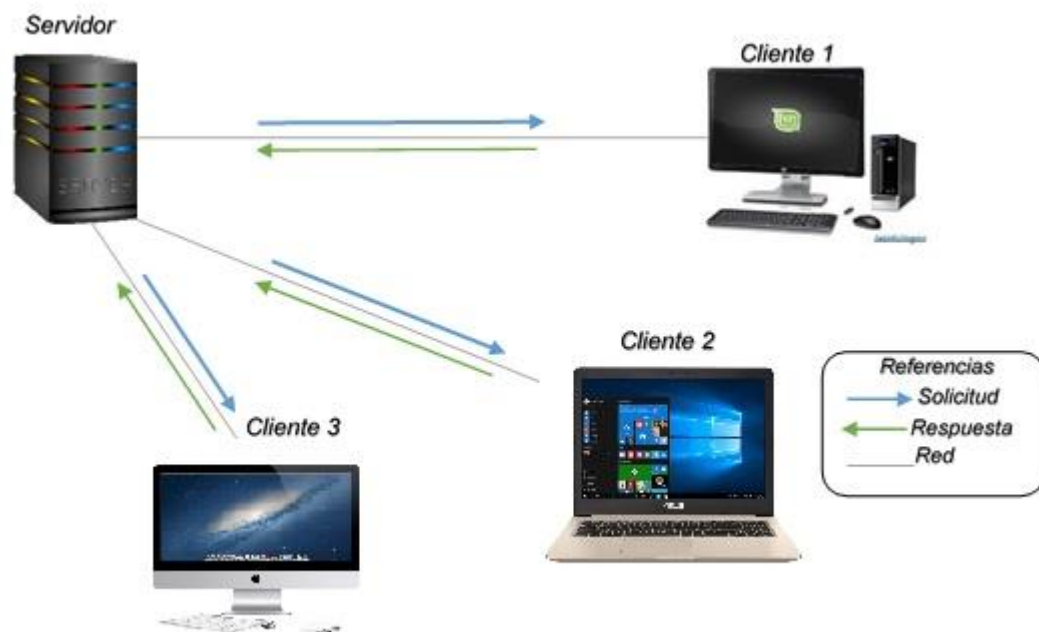
Además, se cuenta con la supervisión por parte del director del proyecto y la colaboración del equipo de trabajo CALUMET, agentes de gran apoyo en la realización de este proyecto.

### 3. MARCO TEÓRICO

#### 3.1 ARQUITECTURA CLIENTE/SERVIDOR

En la arquitectura cliente/servidor cada uno de los clientes produce un mensaje solicitando un determinado servicio a un servidor (hace una petición) estos envían uno o varios mensajes como respuesta (responden peticiones o provee un servicio). La mayoría del trabajo pesado (procesos de base de datos, procesar la lógica de la aplicación entre otros) está a cargo de los servidores, el cliente comúnmente se encarga de las funciones de administración de la interfaz de usuario, interacción con el usuario, recibir resultados del servidor, generar requerimientos de base de datos, entre otros.

Figura 1 Modelo Cliente/servidor



**3.1.1 Características de la arquitectura Cliente/Servidor.** Las características básicas de una arquitectura Cliente / Servidor son:

- Es quien inicia solicitudes o peticiones, tienen por tanto un papel activo en la comunicación.
- El proceso del cliente da la interface entre usuarios y el resto del sistema, maneja recursos compartidos tales como bases de datos, impresoras, módems, etc.
- El cliente y el servidor pueden actuar como una sola entidad y también pueden actuar como entidades separadas, realizando actividades independientes.
- Las tareas del cliente y el servidor tienen diferentes requerimientos como: velocidad del procesador, memoria o capacidad del disco, por tanto, la plataforma de hardware y el sistema operativo del cliente y del servidor no son siempre la misma y eso se conoce como ambiente heterogéneo.
- La escalabilidad horizontal permite agregar más estaciones de trabajo activas sin afectar el rendimiento y la escalabilidad vertical permite mejorar las características del servidor o agregar múltiples servidores.

**3.1.2 Clasificación de las arquitecturas Cliente/Servidor.** Los sistemas Cliente/Servidor se clasifican de acuerdo con el nivel de abstracción del servicio que se ofrece. Se distinguen tres componentes básicos de software para la clasificación:

- **Presentación:** Presentación de resultados al usuario de forma comprensible.
- **Lógica de aplicación:** Esta capa es la responsable del procesamiento de la información que tiene lugar en la aplicación.
- **Base de datos:** Está compuesta por los archivos que contienen los datos persistentes de la aplicación.

La siguiente es la clasificación de la arquitectura Cliente/Servidor:

**3.1.2.4 Arquitectura Cliente/Servidor de dos capas.** Consiste en una capa de presentación y lógica de la aplicación; y otra de la base de datos, cuando el cliente solicita recursos entonces el servidor responde directamente a la solicitud con sus propios recursos. Normalmente esta arquitectura es utilizada en las siguientes situaciones:

- Cuando se requiere poco procesamiento de datos en la organización.
- Cuando se tiene una base de datos centralizada en un solo servidor.
- Cuando la base de datos es relativamente estática.
- Cuando se requiere un mantenimiento mínimo.

**3.1.2.5 Arquitectura Cliente/Servidor de tres capas.** Define como organizar el modelo de diseño en capas, que pueden estar físicamente distribuidas, es decir que los componentes de una capa solo pueden hacer referencia a componentes en capas inferiores. Este patrón es importante porque simplifica la comprensión y la organización del desarrollo de sistemas complejos, reduciendo las dependencias de forma que las capas más bajas no conscientes de ningún detalle o interfaz de las superiores, está compuesta de:

- Un equipo cliente con una interfaz de usuario (habitualmente se utiliza un navegador web), que solicita los recursos.
- El servidor de aplicaciones (o software intermedio), cuya tarea es prestar los recursos solicitados, pero que requiere de otro servidor para hacerlo.
- El servidor de datos que almacena y proporciona al servidor de aplicaciones los datos que requiere.

**3.1.2.6 Arquitectura Cliente/Servidor aplicada.** En el desarrollo de este proyecto se recurre a arquitectura de tres capas, debido a las ventajas ofrecidas como: Escalabilidad, fácil mantenimiento y el manejo de un mayor número de usuarios que la ofrecida por la arquitectura C/S de dos capas. La arquitectura es aplicada de la siguiente forma:

- Capa de Cliente: Interfaz con el usuario, se usa un navegador web.

- **Capa Intermedia:** Para los servicios del negocio se utiliza un computador configurado como servidor web, el cual almacena el portal web conformado por páginas JSP y JavaBeans. Allí se realizan los procesos complejos, y se solicitan los servicios del servidor de datos cuando es necesario acceder a la información almacenada en la base de datos.
- **Capa de Servidor:** Se utiliza el motor de bases de datos MySQL, el cual se encuentra en el mismo servidor web.

### **3.1.3 Ventajas del esquema Cliente/Servidor**

- La existencia de plataformas de software y hardware de varios fabricantes y cada vez más económicas contribuye a la reducción de costos y favorece la flexibilidad en la implantación y actualización de soluciones.
- Este esquema facilita la integración entre sistemas heterogéneos y comparte información permitiendo que las máquinas existentes puedan ser utilizadas con interfaces amigables al usuario, de esta forma integrar los computadores con sistemas medianos y grandes, sin necesidad de que todos tengan que utilizar el mismo sistema operacional.
- Facilita a los diferentes departamentos de una organización soluciones locales, permitiendo la integración de la información principal totalmente.

### **3.1.3 Desventajas del esquema Cliente/Servidor**

- El mantenimiento de los sistemas es complejo pues implica la interacción de diferentes partes hardware y software de diferentes proveedores, lo cual dificulta el diagnóstico de fallas.
- Se cuenta con escasas herramientas para la administración y ajuste del desempeño de los sistemas, además se deben tener estrategias para el manejo de errores y para salvaguardar la consistencia de los datos.
- La seguridad del esquema C/S es preocupante, un ejemplo: las validaciones y verificaciones que se deben hacer tanto en el cliente como en el servidor.

- El desempeño es un aspecto a tener en cuenta en el esquema C/S, problemas de este estilo pueden presentarse por congestión en la red.

## **3.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS**

Las páginas dinámicas aportan grandes beneficios porque permiten entrar a bases de datos para extraer información que pueda presentarse al usuario, dependiendo de algunos permisos y de la misma forma para almacenar información.

Existen diferentes tecnologías para el desarrollo de páginas dinámicas entre ellas están:

**3.2.1 Código del Lado del Cliente (Client Side Scripts).** Código ejecutado por los navegadores, el cual los computadores clientes tienen instalados. Las tecnologías más comunes de este tipo son:

- JavaScript: Lenguaje de programación interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C. Permite, crear ventanas, mostrar texto en movimiento y verificar las entradas a un formulario.
- Controles Activos: tecnología Microsoft que permite la creación de aplicaciones Windows, como pueden ser Visual Basic Script o Visual C. Es la respuesta de Microsoft a los Applets de Java.
- Java Applets: Programas escritos en lenguaje de programación Java, se incrustan en HTML y se ejecutan en el navegador gracias a la Máquina Virtual de Java (JVM) que lleva éste incorporado.

**3.2.2 Código del Lado del Servidor (Server Side Scripts).** Código que se ejecuta en el servidor. Para su actividad el programa ejecuta y procesa los datos o peticiones que el usuario envía desde su navegador, para luego enviar los

resultados del programa en una página HTML que el usuario verá normalmente en su navegador. Los más usados son:

- ASP (Active Server Pages): Permite crear dinámicamente páginas Web mediante HTML, scripts, y componentes de servidor ActiveX reutilizables, requiere de un computador configurado como Servidor Web de Microsoft (Microsoft WebServer), el navegador del cliente es indiferente pues el trabajo se realiza del lado del servidor. Da gran uso en la gestión de Bases de Datos ya que puede conectarse a SQL, Access, Oracle u otras.
- PHP (PHP Hypertext Pre-processor): Lenguaje de programación interpretado, diseñado para la creación de páginas web dinámicas. Es un lenguaje de código abierto (Open Source) y gratuito. Su gran potencia se encuentra en la interacción con los motores de bases de datos como Oracle y MySQL.
- JSP (Java Server Pages): tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo. Permiten la utilización de código Java mediante scripts.

**3.2.3 Tecnología aplicada.** La tecnología aplicada para la creación del portal web fue JSP, por lo tanto, los nuevos servicios son desarrollados con esta misma tecnología, ya que permite producir aplicaciones independientes de la plataforma y portables a otros sistemas operativos y servidores web. Una página JSP se compila a una aplicación Java la primera vez que se invoca, y de esta aplicación Java se crea una clase que empieza a ejecutarse en el servidor como un servlet. Un JSP es una página web con etiquetas especiales y código Java incrustado, mientras que un servlet es un programa que recibe peticiones y genera a partir de una de ellas una página Web.

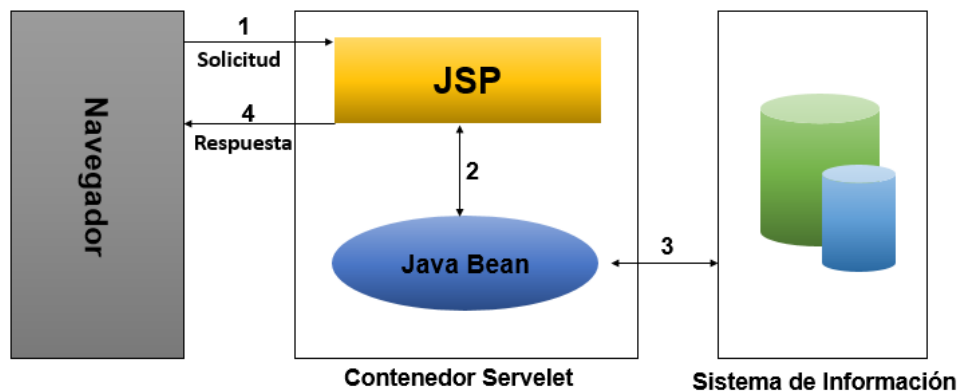
#### **3.2.3.1 Modelo de acceso a JSP**

- Un usuario en su navegador web cliente hace una petición que es enviada a un

archivo JSP. Este archivo accede a componentes del servidor que generan contenido dinámico y lo presentan en el navegador.

- Después de recibir la petición del cliente, el archivo JSP pide información de un JavaBean si es necesario.
- El JavaBean en turnos puede pedir información de otro JavaBean o de una base de datos.
- Una vez el JavaBean genera el contenido, el archivo JSP puede consultar y presentar el contenido del JavaBean al navegador.

Figura 2 Modelo de acceso a JSP



La primera vez que un archivo JSP es invocado, este es compilado en un objeto, la respuesta del objeto es HTML estándar, el cual es interpretado por el navegador para ser presentado al usuario. Después de la compilación, el objeto de la página es almacenado en la memoria del servidor. En las peticiones posteriores a esta página, el servidor revisa si el archivo JSP ha cambiado. Si no ha cambiado, el servidor utiliza el objeto de la página compilada guardado en memoria para generar la respuesta al cliente, en caso contrario el servidor automáticamente compila el archivo de la página y reemplaza el objeto en la memoria.

### 3.3 BASES DE DATOS

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso, con una redundancia controlada y una estructura que refleja las interrelaciones y restricciones existentes en el mundo real. En la base de datos se almacena información considerada necesaria para una determinada organización o negocio. Existen modelos que describen la estructura de una base de datos (entidades, atributos y relaciones), la mayoría de los modelos de datos poseen un conjunto de operaciones básicas como consultar, actualizar y eliminar.

**3.3.1 Modelos de Bases de Datos.** Las bases de datos se clasifican de acuerdo con su modelo de administración de datos. Algunos modelos utilizados con frecuencia son:

- **Base de Datos Jerárquica.** Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento. Está limitado por su incapacidad de representar eficientemente la redundancia de datos.
- **Base de Datos de Red.** En este modelo se permite que un mismo nodo tenga varios padres. Ofrece una solución eficiente al problema de redundancia de datos; sin embargo, la dificultad para administrar los datos en una base de datos de red ha conllevado a que sea un modelo usado más por programadores que por usuarios finales.
- **Base de Datos Relacional.** Es el más utilizado para modelar problemas reales y administrar datos dinámicamente. Su fundamento es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos, también llamados tuplas. Cada relación es una tabla que está compuesta por

registros (las filas de una tabla), que representan las tuplas, y campos (las columnas de una tabla). Los datos pueden ser recuperados o almacenados mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

**3.3.2 Manejadores o Gestores de Bases de Datos.** El sistema manejador de bases de datos es la porción más importante del software de un sistema de base de datos. Un DBMS es una colección de numerosas rutinas de software interrelacionadas, cada una de las cuales es responsable de alguna tarea específica.

Las funciones principales de un DBMS son:

- Crear y organizar la Base de Datos.
- Establecer y mantener las trayectorias de acceso a la base de datos de tal forma que los datos puedan ser capturados rápidamente.
- Manejar los datos de acuerdo a las peticiones de los usuarios.
- Registrar el uso de las bases de datos.
- Interacción con el manejador de archivos a través de las sentencias en Lenguaje Manipulador de Datos (Data Manipulation Language, DML) al comando del sistema de archivos.
- Respaldo y recuperación: Consiste en contar con mecanismos implantados que permitan la recuperación fácilmente de los datos en caso de ocurrir fallas en el sistema de base de datos.
- Control de concurrencia: consiste en controlar la interacción entre los usuarios concurrentes para preservar la consistencia de los datos.
- Seguridad e Integridad: consiste en contar con mecanismos que permitan el control de la consistencia de los datos evitando que estos se vean perjudicados por cambios no autorizados o previstos.

**3.3.3 MySQL.** Es un sistema de base de datos operacional considerado uno de los más importantes, utilizado por usuarios del medio para el diseño y programación de base de datos de tipo relacional. Además, MySQL es conocida por desarrollar alta velocidad de búsqueda de datos e información, a diferencia de sistemas anteriores.

#### **3.3.4 Ventajas de MySQL**

- El MySQL es un Open Source, o sea código abierto que puede ser usado y modificado.
- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una maquina con escasos recursos sin ningún problema.
- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
- Su conectividad, velocidad, y seguridad hacen de MySQL altamente apropiado para acceder a bases de datos en internet.

#### **3.4 NETBEANS**

Es un entorno de desarrollo integrado (IDE), siendo una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas escritos en JAVA, pero puede servir para cualquier otro tipo lenguaje de programación. NetBeans es un producto libre y gratuito sin restricciones de uso. El NetBeans es un entorno de desarrollo integrado de código abierto escrito completamente en Java usando la plataforma NetBeans, soporta desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles).

La versión actual es NetBeans IDE 8.2, desde NetBeans IDE 6.5 se extienden las características existentes del Java EE (incluyendo Soporte a Persistencia, EEJB 3 y JAX-WS).

Adicionalmente, el NetBeans Enterprise Pack soporta el desarrollo de Aplicaciones empresariales java EE 5, incluyendo herramientas de desarrollo visuales de SOA, herramientas de esquemas XML, orientación a web servicios (for BPEL), y modelado UML.

Todas las funciones del IDE son provistas por módulos. Cada Módulo provee una función bien definida, tales como el soporte de Java, edición, o soporte para el sistema de control de versiones. NetBeans contiene todos los módulos necesarios para el desarrollo de aplicaciones Java en una sola descarga, permite al usuario comenzar a trabajar inmediatamente.

### **3.5 SISTEMA DE CONTROL DE VERSIONES**

Un sistema de control de versiones es un software que administra el acceso a un conjunto de ficheros, y mantiene un historial de cambios realizados. El control de versiones es útil para guardar cualquier documento que cambie con frecuencia, o el código fuente de un programa.

Normalmente consiste en una copia maestra en un repositorio central, y un programa cliente con el que cada usuario sincroniza su copia local. Además, el repositorio guarda registro de los cambios realizados por cada usuario, y permite volver a un estado anterior en caso de necesidad.

**3.5.1 Subversión.** Esto permite recuperar versiones antiguas de los datos o examinar cómo han ido evolucionando esto. Su objetivo es ser un sucesor prácticamente compatible del ampliamente usado Concurrent Version system (CVS). Subversión puede trabajar a través de redes, lo que permite que las personas que estén en diferentes computadores puedan usarlo, con la posibilidad de que varias personas modifiquen y gestionen el mismo conjunto de datos desde sus sitios que promueve la colaboración, y como el trabajo está versionado, ya que, si se produce algún cambio incorrecto de los datos, sólo hace falta deshacerlo.

### 3.6 PROGRAMACIÓN UTILIZADA

Para el desarrollo de este proyecto se usó la Programación Orientada a Objetos (P.O.O.). La P.O.O. es una de las formas más populares de programas que usa objetos y sus interacciones para diseñar aplicaciones y programas de computador, intenta simular el mundo real a través del significado de objetos que contienen características y funciones; abstrae algunas características de sistemas naturales complejos como son:

- Atributos: estado del objeto.
- Métodos: comportamiento del objeto.
- Herencia: comportamientos comunes entre objetos relacionados para hallar relaciones de especialización y generalización de comportamientos.

**3.6.1 Clases.** Definición de todos los elementos de que este hecho un objeto. Cuando se programa un objeto y se definen sus características y funcionalidades, realmente se programa una clase. Por lo tanto, para realizar la abstracción de sistemas naturales, observamos y analizamos un grupo de cosas con características comunes, el resultado de esta abstracción será válido para todas estas cosas.

**3.6.2 Objetos.** Cualquier cosa real o abstracta, que posee atributos y un conjunto de operaciones que manipulan esos atributos que da un comportamiento particular. Un objeto es una instancia de una clase, el estado del objeto se determina por el estado (valor) de sus propiedades o características (atributos).

**3.6.3 Atributos.** Características de un objeto siendo un conjunto de datos (valores) y calificadores para aquellos datos. Estos atributos pueden ser desde tipos de datos simples (enteros, caracteres, cadenas de texto) hasta otros objetos.

**3.6.4 Métodos.** Son funciones o procedimientos propios de la clase que pueden tener acceso a los atributos de la misma para realizar las operaciones para los que son programados.

**3.6.5 Herencia.** Se fundamenta en usar una clase ya creada para tomar sus características en clases más especializadas o derivadas de ésta para reutilizar el código que sea común con la clase base, y solamente definir nuevos métodos o redefinir algunos de los existentes para ajustarse al comportamiento particular de esta subclase.

### **3.6.6 Beneficios de la Programación Orientada a Objetos**

- Permite obtener aplicaciones modificables y fácilmente extensibles a partir de componentes reutilizables.
- Disminución en el tiempo de desarrollo gracias a la reutilización del código.
- El desarrollo del software es más intuitivo porque las personas piensan naturalmente en términos de objetos más que en términos de algoritmos de software.

A continuación, se presenta una breve descripción de Java, el lenguaje de programación orientado a objetos que se usó en el desarrollo de este proyecto:

**3.6.7 Java y JDK (Java Development Kit).** Java es un lenguaje desarrollado por Sun Microsystems, en el año 2009 fue adquirida por la compañía Oracle.

El JDK es un software que provee herramientas de desarrollo para la creación de programas en java.

Para trabajar con Java se necesita un kit de desarrollo que proporciona:

- Un compilador: `javac`
- Un intérprete: `java`.
- Un generador de documentación: `javadoc`

**3.6.7.1** Un visor de applet para generar sus vistas previas, ya que un applet carece de método main y no se puede ejecutar con el programa java: Appletviewer.

## **3.7 SERVIDORES WEB**

Es un tipo de software que se encuentra a la espera de una petición hecha por una aplicación cliente y da respuesta a dicha petición a través de una página web. Para cada transacción el servidor debe realizar dos acciones básicas: integrar todos los componentes de la página (texto, imágenes, vídeo, scripts, etc.) y enviarla rápidamente al usuario. A continuación, se describe el servidor Web que se ajusta a la tecnología escogida para el proyecto.

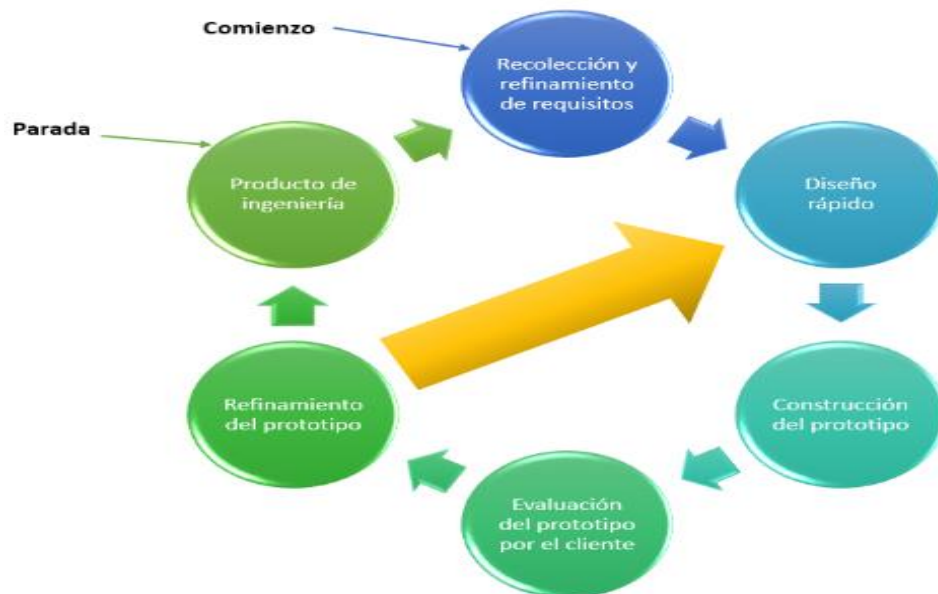
**3.7.1 Servidor Apache Tomcat.** Servidor de aplicaciones Java basado en los estándares definidos por Sun Microsystems. Tomcat es desarrollado como parte del proyecto de código abierto de la fundación de software Apache y es uno de los servidores de aplicaciones Java más utilizados, en especial porque es liviano, cumple con todos los estándares, sencillo de instalar, tiene muy buena documentación y es gratuito, además por ser escrito en Java funciona en cualquier sistema operativo que disponga de la Máquina Virtual de Java (JVM).

## 4 MARCO METODOLÓGICO

### 4.1 PROTOTIPO EVOLUTIVO

Para realizar los nuevos servicios para los portales web comunidad académica de las diferentes escuelas y facultades de la Universidad Industrial de Santander se propone como metodología de desarrollo el prototipo evolutivo.

Figura 3 Prototipo evolutivo



La elección de la metodología de prototipo evolutivo se debe a las siguientes razones:

- Es deseable tener un bosquejo de lo que se desee mejorar o crear para poder incorporar sugerencias de cambio por parte de los usuarios del portal de las escuelas en las etapas tempranas del desarrollo.

- Por otra parte, es necesario saber lo antes posible si hemos interpretado correctamente las especificaciones y las necesidades de las escuelas y de los profesores.
- En muchos casos los usuarios no tienen una idea definida de lo que desean, por lo tanto, debemos tomar decisiones y suponer qué es lo que el usuario quiere.
- Por este motivo, la emisión de los prototipos brinda la posibilidad de efectuar refinamientos de los requerimientos en forma sucesiva a fin de acercarse al producto deseado. Con el prototipo evolutivo se comienza diseñando y construyendo las partes más importantes de la aplicación en un prototipo que posteriormente se refina y ampliará hasta que el prototipo se termine. Este prototipo será el software que se entregará al final.
- La decisión se fundamenta en la ventaja de la realización de los cambios en etapas tempranas y la posibilidad de emisión de varios prototipos evaluables durante el desarrollo, obteniéndose de este modo, y de forma paralela, una metodología integral también para el proceso de evaluación del programa.
- Esta metodología propicia un intercambio de conocimientos y de autocrítica al sistema, lo que conlleva a que se produzcan muchas pruebas antes de liberar una nueva versión, así como mejoras rápidas a problemas que puedan surgir durante su uso.

Procedimiento a seguir para la metodología planteada:

- Hacer un análisis de los requerimientos para la construcción de los prototipos.

- Desglosar los objetivos globales con el fin de tener una idea más detallada del software a realizar, mediante reuniones entre los desarrolladores y los usuarios, en las cuales se identifican los requerimientos de los usuarios y se concluyen los aspectos que requieren una mayor definición.
- Presentar al usuario el diseño de un prototipo enfocado en los aspectos visuales del software, métodos de entrada y formatos de salida, para proceder a la construcción del mismo.
- Evaluación del prototipo por parte del usuario para filtrar los requisitos del software a desarrollar.
- Se produce un proceso interactivo en el que el prototipo es depurado para satisfacer necesidades del usuario, de igual forma el desarrollador obtiene una mejor comprensión de lo que hay que hacer para la entrega del producto final de ingeniería requerido por el usuario.

## **4.2 LENGUAJE DE MODELADO UNIFICADO**

El Lenguaje de Modelado Unificado o Unified Modeling Language (UML), es el más utilizado en la actualidad. Es un lenguaje gráfico estándar para visualizar, especificar, construir y documentar un sistema para describir un modelo del sistema, incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables. UML no es un método de desarrollo porque no indica los pasos a seguir para llegar al código, es decir, no especifica como pasar del análisis al diseño y de este al código. Al no ser un método de desarrollo resulta ser independiente del ciclo de desarrollo que se siga, puede encajar en un ciclo en cascada, evolutivo, espiral o en métodos ágiles de desarrollo.

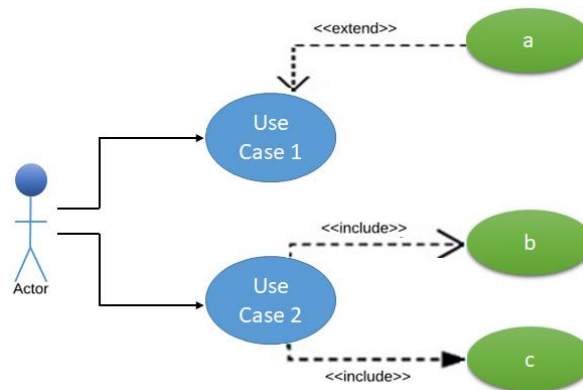
**4.2.1 Diagramas de UML.** Los diagramas UML utilizados en el desarrollo de este proyecto fueron diagramas de casos de uso y diagramas de secuencias.

Las principales razones por las cuales se prefirió UML como el lenguaje de modelado son:

- UML tiene una notación gráfica muy expresiva que permite representar todas las fases de un proyecto informático, desde el análisis con casos de uso, el diseño con diagramas de clases, objetos, etc.
- UML facilita el entendimiento de la información, la función y el comportamiento de un sistema, haciendo fácil el análisis de los requerimientos, ya que sirve de apoyo en los procesos de análisis de un problema.
- UML permite a los creadores de sistemas realizar diseños que faciliten la comunicación a otras personas de manera convencional.
- UML permite generar un punto de comparación entre lo logrado y lo planificado.

**4.2.2 Diagramas de casos de uso.** Representación gráfica del entorno del sistema (actores) y su funcionalidad principal. Describe lo que hace el sistema desde el punto de vista de un observador externo, concentrándose en expresar lo que hace el sistema y no en dar respuesta de cómo lograr su comportamiento.

Figura 4 Diagramas de Casos de Uso



**4.2.2.1 Actores.** Un actor en un caso de uso representa un rol, que alguien o algo puede desempeñar dentro un sistema y no un alguien o algo específico. En este proyecto se destacan tres clases de actores:

**Administradores:** Son usuarios que además de pertenecer a la categoría de usuarios tienen un perfil de administrador, con el cual tiene permisos extras a los que tiene un usuario comúnmente dentro del sitio; algunos de estos son los auxiliares de administración del portal, los profesores, las secretarías de las escuelas, entre otras. Dentro de esta categoría se incluye también el súper administrador.

**Súper Administrador:** Es el usuario que puede administrar, controlar y modificar los portales web de las escuelas, sus parámetros y sus usuarios.

**Usuario Portal Web Comunidad Académica:** Es el tipo de usuario común de los portales web y a quien van dirigidas las páginas de servicio. Este usuario solo tiene control sobre sus servicios permitidos.

**4.2.2.2 Inclusión (include).** Es una forma de interacción, un caso de uso dado puede "incluir" otro. Una inclusión es utilizada para indicar que un caso de uso depende de otro, es decir, la funcionalidad de determinado caso de uso se requiere para realizar las tareas de otro. En la figura 4 el caso de uso "Caso de uso 1" depende de los casos de uso "A" y "B".

**4.2.2.3 Extensión (extend).** Es otra forma de interacción, una extensión representa una variación de un caso de uso a otro, es decir, una dependencia específica entre los casos de uso, a través de la cual un caso de uso puede extender a otro.

**4.2.2.4 Diagramas de secuencias.** Es aquel que muestra la forma en que los objetos interactúan entre sí al transcurrir el tiempo. Consta de objetos que se representan del modo usual: rectángulos con nombre (subrayado), mensajes representados por líneas continuas con una punta de flecha y el tiempo

representado como una progresión vertical.

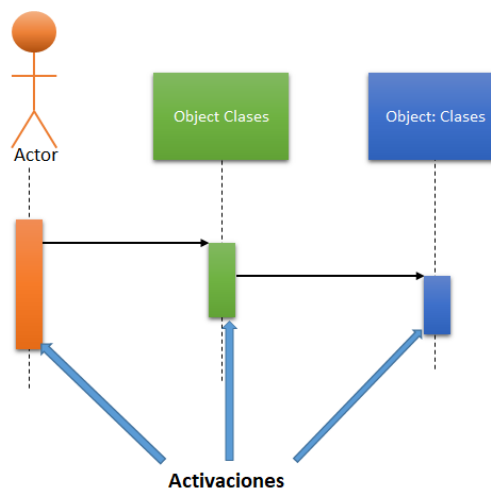
**Objetos:** Se ubican en la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen al diagrama. La línea que está debajo de cada objeto será una línea discontinua conocida como la línea de vida de un objeto. Con la línea de vida se encuentra un pequeño rectángulo conocido como activación, el cual representa la ejecución de una operación que realiza el objeto.

**Mensaje:** Un mensaje que va de un objeto a otro pasa la línea de vida de un objeto a otro. Un objeto puede enviarse un mensaje a sí mismo. Un mensaje puede ser simple, sincrónico o asincrónico.

**Tiempo:** El diagrama representa al tiempo en dirección vertical. Inicia en la parte superior y avanza hacia la parte inferior. Un mensaje que esté más cerca de la parte superior ocurrirá antes que uno que esté cerca de la parte inferior.

**GUI:** (Siglas en Ingles) La interfaz gráfica de usuario; es la interfaz de interacción del usuario y en la que más interactividades se presentan con otros objetos.

Figura 5 Diagrama de Secuencias



## 4.3 ESTÁNDARES DE PROGRAMACIÓN

**4.3.2 Modelo de datos.** Es un lenguaje utilizado para la descripción de una base de datos, por lo general permite describir estructuras de datos de la base de datos (el tipo de datos que incluye la base y la forma en que se relacionan), las restricciones de integridad (las condiciones que los datos deben cumplir para reflejar correctamente la realidad deseada) y las operaciones de manipulación de los datos (agregar, borrar, modificar).

**4.3.3 Nombres de las tablas.** Los nombres de los campos, así como de las tablas de la base de datos, se escriben en minúsculas, exceptuando los prefijos TP, TR y TB que indican si la tabla es principal, relacional o básica, respectivamente, y exceptuando también la primera letra de cada palabra que conforme su nombre; si es un nombre compuesto por dos o más palabras, los nombres tendrán en mayúscula la primera letra de cada palabra que la forma.

Como se mencionó, se han definido tres categorías para las diferentes tablas que conforman la base de datos. Dada la categoría de la tabla, se agrega un prefijo a su nombre que permita conocer la categoría a la que pertenece. Las categorías son:

**4.3.3.1** Tabla básica: Aquella cuyos registros son necesarios para el correcto funcionamiento de la base de datos. Estas tablas no experimentan muchos cambios en los datos. El prefijo a anteponer a los nombres de estas tablas es “TB\_”, es decir la tabla que almacena las categorías de clasificación de los usuarios del portal web es llamada “TB\_Categorias”, por ejemplo.

**4.3.3.2** Tabla de Relación: Surge de la relación muchos a muchos de una o dos tablas cualquiera. Los nombres de las tablas de relación deben ser siempre descriptivos para cada relación. El prefijo a anteponer a los nombres de estas tablas es “TR\_”, por ejemplo, la tabla “MiembrosConsejo” es conocida como “TR\_MiembrosConsejo”.

**4.3.3.3** Tabla Principal: Aquella cuyo número de registros tiende a crecer en gran cantidad y que además no es posible clasificar como tabla básica o de relación. Un ejemplo de tabla principal es la tabla que almacena los usuarios del portal EISIWeb. El prefijo a anteponer a los nombres de estas tablas es “TP\_”, es decir la tabla “Usuarios”, es conocida como “TP\_ Usuarios”.

**4.3.4 Clases.** Los nombres de las clases deben ser sustantivos en plural, la primera letra de cada palabra debe ser mayúscula. Estos deben ser simples, descriptivos como, por ejemplo: TipoPlan.java, Usuarios.java, NotasCartelera.java, etc.

**4.3.5 Páginas JSP.** Los nombres de las páginas JSP que componen los portales web comunidad académica son escritos de manera que la primera letra es una mayúscula seguido de letras minúsculas, en caso de que el nombre del JSP sea compuesto por dos o más palabras, entonces la primera de cada palabra debe ir en mayúscula, por ejemplo: ListarTemaPlan.jsp, CrearPreguntaProfesor.jsp, PeticionesBancoPregunta.jsp, etc.

**4.3.6 Organización de Directorios.** Los directorios del sitio están organizados de tal manera que los archivos que se almacenen en ellos correspondan a lo que describe el nombre del directorio. Por ejemplo: Aula.

- El sitio cuenta con un directorio llamado “images”; en éste se encuentran almacenados todos los archivos .jpg, .gif, .png.
- Para el desarrollo de este proyecto se crearon los directorios “/eisi/PerfilProfesores”, “/eisi/images/BlogProfesor”, los cuales almacenan los archivos correspondientes al funcionamiento de estos servicios

## **5 DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO**

Para el desarrollo de este proyecto se siguió la metodología de prototipo evolutivo. Al iniciar el proyecto se elaboró un primer prototipo durante la fase de requerimientos, el cual fue mejorado con la inclusión de nuevos requerimientos surgidos en la fase de desarrollo, a medida que se generaba un prototipo, el mismo era sometido a pruebas de funcionamiento y se le realizaban los refinamientos pertinentes a partir del resultado de dichas pruebas.

### **5.1 PROTOTIPO ESPERADO**

Al iniciar el proyecto aún no se contaba con una concepción clara de lo sería el producto final, sin embargo, en el transcurso del desarrollo y evolución de los prototipos, las pruebas y análisis del sistema se pudo comprobar que se estaba acercando a los requerimientos iniciales, esto con el fin de enfocar el desarrollo a la solución de las necesidades de los usuarios.

El objetivo específico inicial y los requisitos que surgieron se dieron gracias a la realización de prototipos y la realimentación con el cliente. Para cada objetivo se listaron los requerimientos detallados de este, los cuales se cumplieron para el prototipo final.

#### **5.1.1 Nuevo servicio para permitir al Comité de Trabajos de Grado cancelar cualquier proyecto en desarrollo**

##### **Objetivo inicial:**

- En el módulo de trabajos y tesis de grados, permitir al Comité de Trabajos de

Grado cancelar cualquier proyecto.

**Requisitos finales del objetivo:**

- Listar los Trabajos de Grado por las categorías nombre del autor, código del estudiante, título del trabajo de grado y nombre del director del trabajo de grado de acuerdo al estándar Elise.
- Una vez listado los trabajos de grado, se permite la selección de un proyecto para acceder a la información, dejar una observación acerca de la decisión de cancelar el trabajo de grado.
- Al momento de cancelar el proyecto, se verá en el acta la decisión tomada por el comité de trabajos de grado.

**5.1.2 En el módulo de trabajos de grados, permitir a las Escuelas que se oculte a los autores el nombre del profesor evaluador de plan.**

**Objetivo inicial:**

- Se debe permitir a las Escuelas que se oculte a los autores el nombre del profesor evaluador de plan. Esto implica corregir los servicios de correos, buscadores, actas y mi proyecto.

**Requisitos finales del objetivo:**

- A solicitud de las Escuelas de Ingeniería Metalurgia y Ciencia de Materiales, Estudios Industriales y Empresariales e Ingeniería Mecánica, se permita ocultar el nombre del profesor evaluador del plan a los autores.
- Actualizar en el enlace gestionar mis proyectos, la pestaña Mi proyecto, para que se oculte el nombre del profesor evaluador del plan a los autores del trabajo de grado.
- Ocultar el nombre del profesor evaluador del plan a los autores del trabajo de

grado, en el menú de Trabajos de grado, consultas trabajo y la pestaña Buscador Proyectos.

- Corregir las actas de trabajo de grado, para que no aparezca el nombre del profesor evaluador del plan de trabajo de grado.

### **5.1.3 Implementar texto Látex en el banco de preguntas del aula**

#### **Objetivo inicial:**

- Para el banco de preguntas implementar el sistema de composición de texto Látex que permite generar estructuras matemáticas en los enunciados de las preguntas.

#### **Requisitos finales del objetivo:**

- Agregar una librería llamada MathJax que permite incluir las matemáticas en las páginas web.
- Agregar un recuadro de vista previa, en la formulación de la pregunta y en la solución, que permita visualizar el texto resultado que el profesor digita.
- Permitir al estudiante visualizar correctamente las matemáticas introducidas en las preguntas de los talleres y exámenes.

### **5.1.4 Control de sesión en el aula virtual.**

#### **Objetivo inicial:**

- Implementar un sistema de control de sesión en el aula virtual que permita controlar el tiempo y avise al usuario en un momento dado del tiempo restante de sesión para que defina si desea continuar la sesión o cerrarla.

**Requisitos finales del objetivo:**

- Implementar el control de sesión, para que se ejecute después de 15 minutos en los talleres, exámenes y foros del aula.
- El estudiante puede decidir si cierra o continua con la sesión.

**5.1.5 Exportar las notas de las actividades del aula en un archivo con formato csv.****Objetivo inicial:**

- Generar en formato csv un archivo con las notas de las diferentes actividades que se realizan en el aula con el fin de facilitar la exportación de notas del aula virtual al sistema de la UIS.

**Requisitos finales del objetivo:**

- Permite seleccionar la actividad sobre la cual se desea exportar las notas en archivo csv.
- Para generar un archivo csv es necesario introducir el año, el periodo y la nota parcial.
- Al generar un archivo csv, se almacena en el servidor y así el profesor pueda acceder a los archivos generados anteriormente.
- En explorar archivos, se listan los archivos csv generados anteriormente, en una tabla donde podrá descargar el archivo o eliminarlo.

### **5.1.6 Actualización en la construcción y modificación del libro en el aula virtual.**

#### **Objetivo inicial:**

- Realizar la construcción y modificación de la forma en que se muestra los servicios de recursos multimedia empleados en el aula virtual, permitiendo menús a los profesores o encargados, para así mostrarle a los estudiantes, los contenidos objeto de estudio que acompañan las diferentes temáticas del aula.

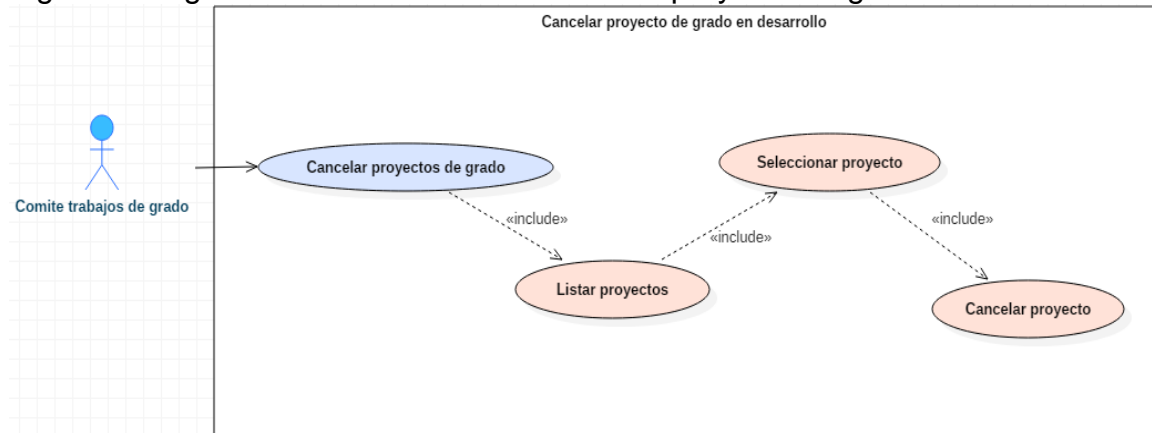
#### **Requisitos finales del objetivo:**

- Mejoras visuales en la construcción y modificación del libro.
- Para construir un libro permite seleccionar una materia existente en el plan de estudios como nombre o digitar un nombre específico y seleccionar un grupo colaborador.
- En una tabla se muestran los libros que tiene el profesor ya creados. También permite gestionar unidades, editar y eliminar el libro.
- En gestionar unidades del libro, las unidades se muestran en una tabla, donde se permite agregar una unidad, gestionar temas, editar y eliminar la unidad.
- En gestionar temas, los temas de la unidad se muestra en una tabla, en la cual permite administrar archivos, agregar enlaces relacionados con el tema, editar y eliminar el tema.
- El administrador de archivos, permite subir un archivo, crear una carpeta, descargar, ver detalles, moverlos a otras carpetas, renombrar el archivo y eliminar.

## 5.2 DIAGRAMAS DE CASOS DE USO

### 5.2.1 Servicio para cancelar cualquier proyecto en desarrollo

Figura 6 Diagrama de casos de uso: Cancelar proyecto de grado en desarrollo.



### 5.2.2 Servicio para implementar Látex en el banco de preguntas del aula virtual

Figura 7 Diagrama de casos de uso: Implementar látex al banco de preguntas del aula virtual.

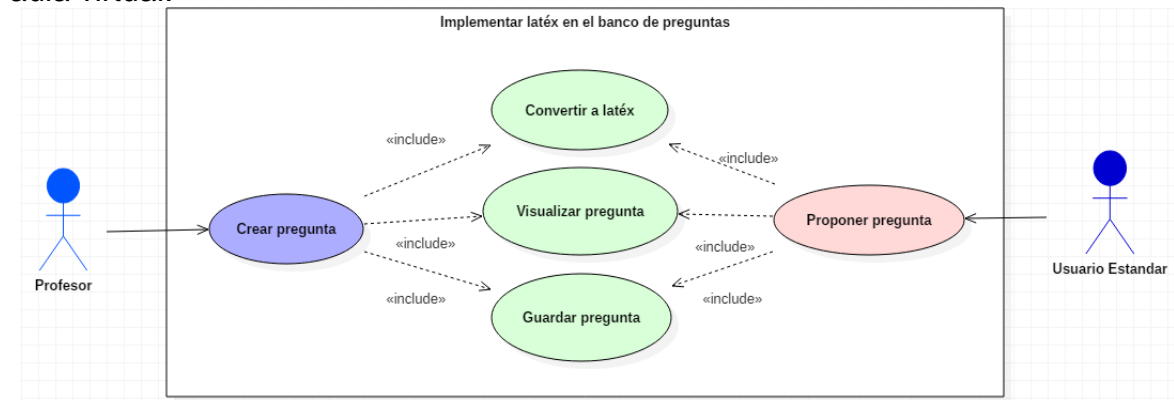


Figura 8 Diagrama de casos de uso: Visualización de preguntas con látex por el estudiante en el aula virtual.

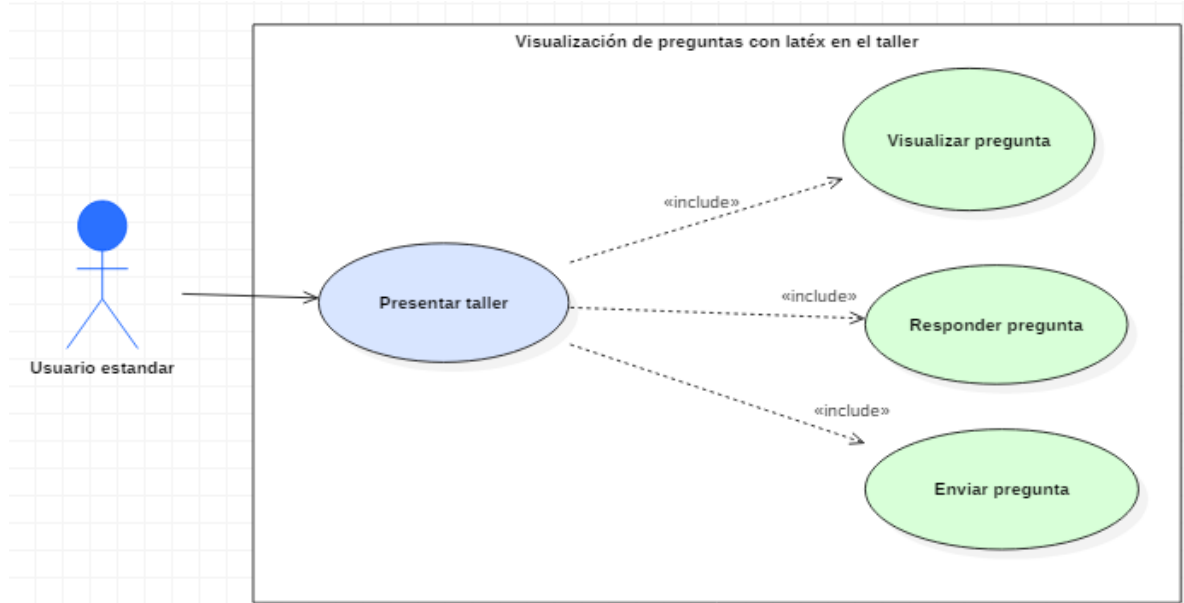
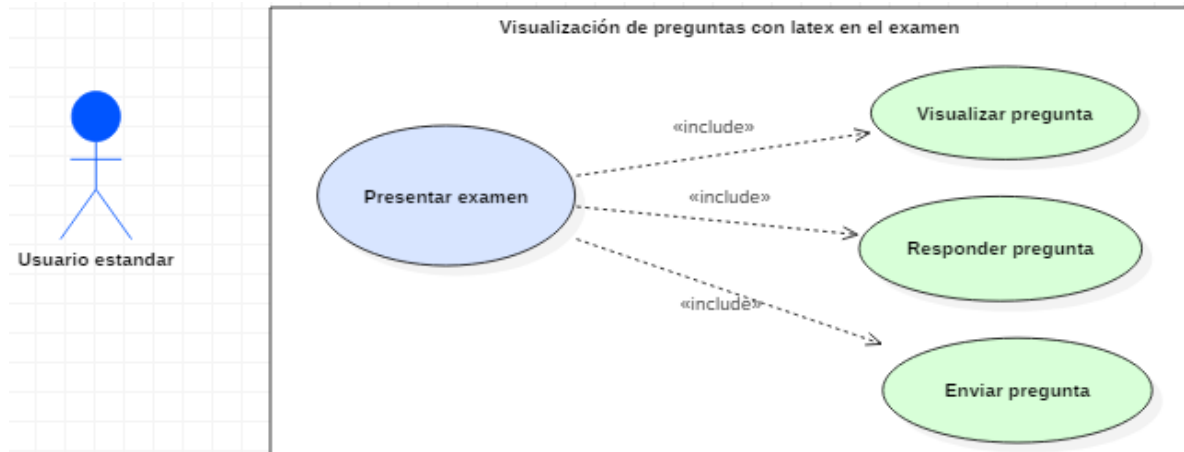
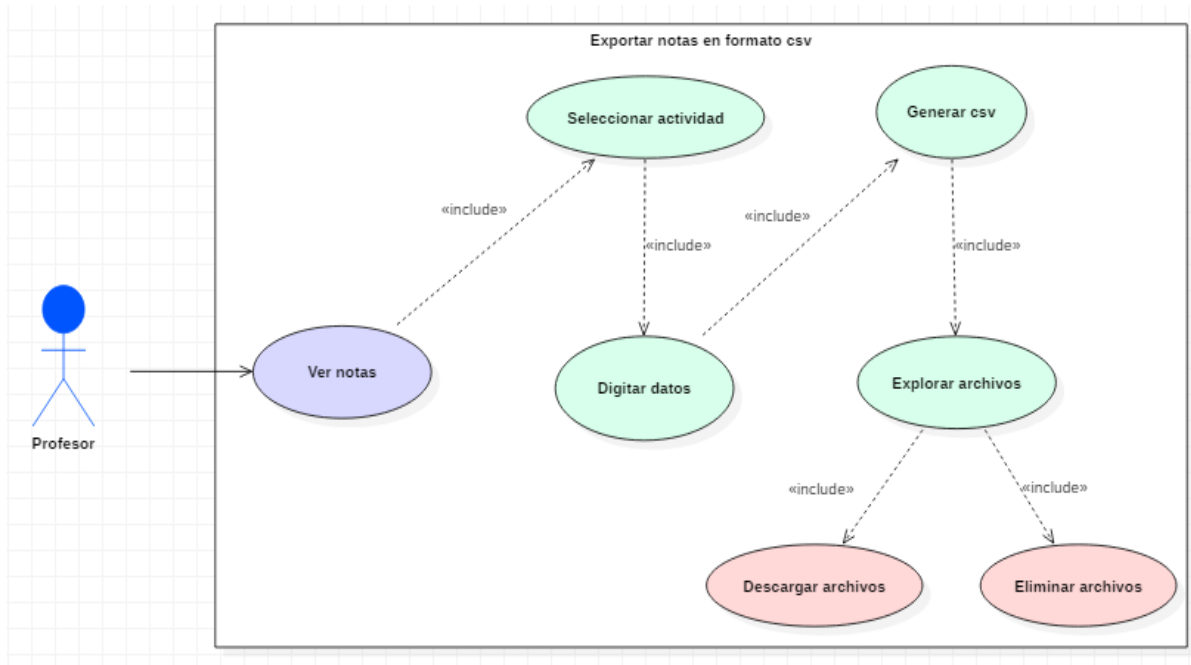


Figura 9 Diagrama de casos de uso: Visualización de preguntas con látex por el estudiante en el aula virtual.



### 5.2.3 Servicio para exportar notas en formato csv

Figura 10 Diagrama de casos de uso: Exportar notas en formato csv del aula virtual



## 5.2.4 Servicio para la construcción del libro del aula virtual

Figura 11 Diagrama de casos de uso: Creación y modificación del libro virtual

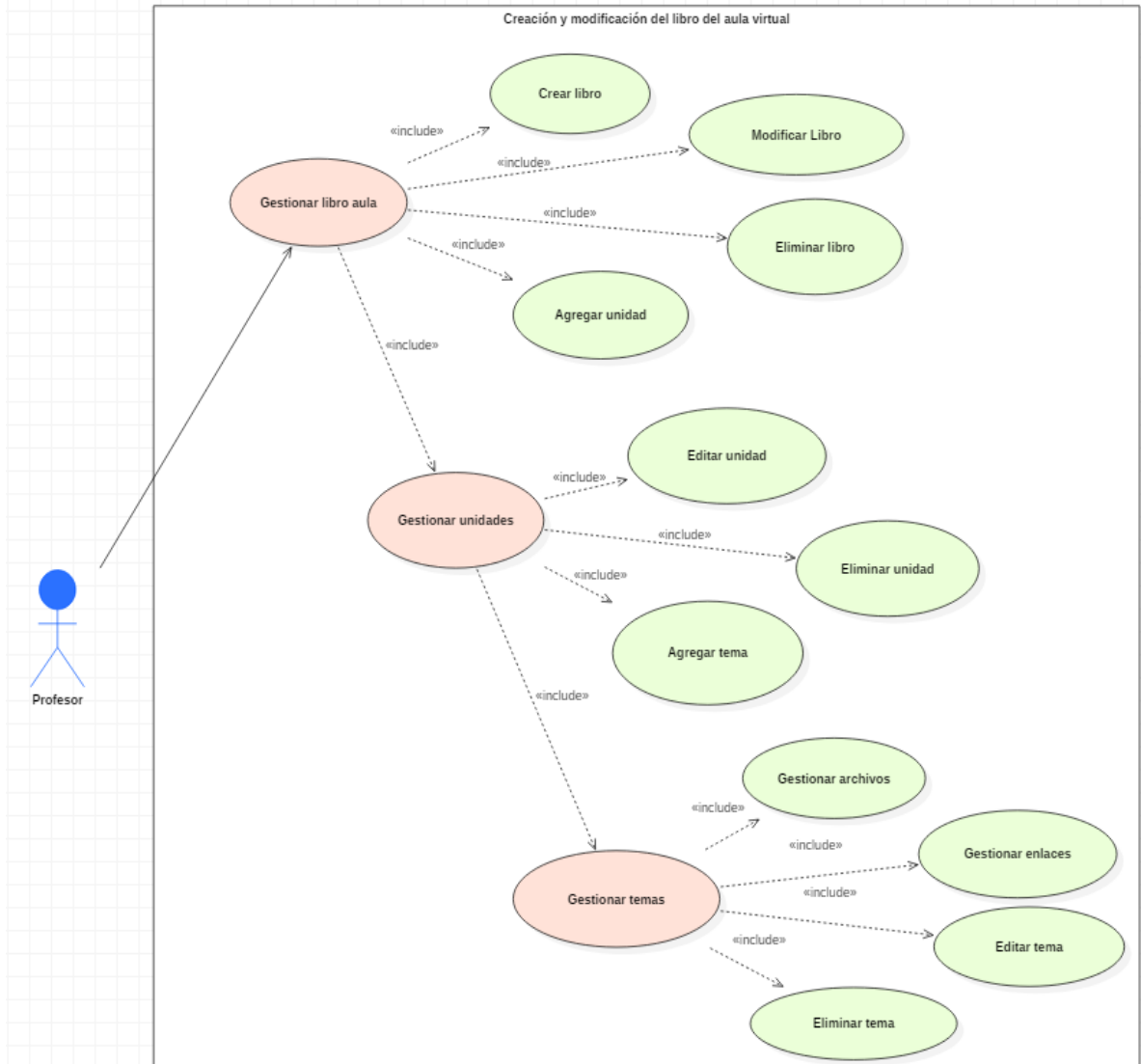
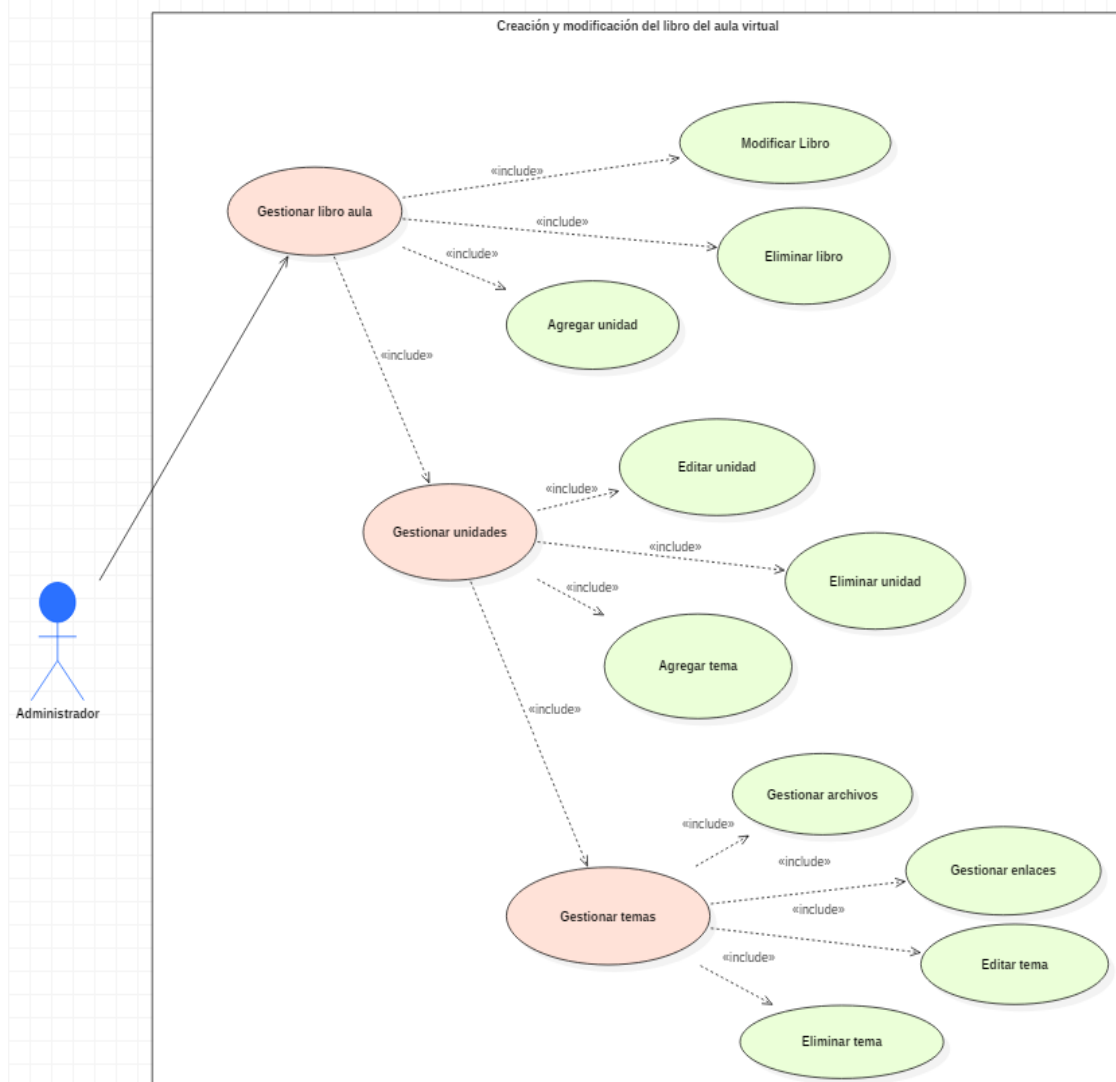


Figura 12 Diagrama de casos de uso: Creación y modificación del libro virtual por un grupo colaborador



### 5.3 DOCUMENTACIÓN DE CASOS DE USO DEL SISTEMA

#### 5.3.1 Servicio para cancelar cualquier proyecto en desarrollo

Tabla 1 Casos de uso: Cancelar proyecto de grado en desarrollo.

TÍTULO	FUNCIONES PRIMARIAS
Cancelar proyectos de grado	Este servicio permite al comité de trabajos de grado listar los proyectos, ver la información del proyecto para dar una observación y cancelar el proyecto definitivamente.
Buscar proyectos	Se buscan los proyectos por nombre del autor, nombre del director de trabajo de grado, título del proyecto y código del estudiante y luego se listan por el criterio seleccionado.
Seleccionar proyecto	Se selecciona el proyecto que se quiere cancelar
Cancelar proyecto	El comité de proyectos de grado puede dar una observación y después cancelar el trabajo de grado.

### 5.3.2 Servicios para implementar $\text{\LaTeX}$ en el banco de preguntas del aula virtual

Tabla 2 Casos de uso: Crear pregunta por un profesor al banco de preguntas del aula virtual.

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Crear pregunta	Este servicio está habilitado para el profesor, permite crear una pregunta ya sea abierta, selección única respuesta, selección múltiple respuesta, verdadero-falso y pregunta con expresiones.
Convertir $\text{\LaTeX}$	Mediante la librería de javascript MathJax permite modelar el texto en ecuaciones matemáticas.
Visualizar pregunta	Permite visualizar las matemáticas que se esté digitando en la pregunta y en la solución.
Guardar pregunta	Guarda las preguntas y soluciones en base de datos en formato $\text{\LaTeX}$ .

Tabla 3 Casos de uso: Proponer pregunta por un usuario al banco de preguntas del aula virtual.

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Proponer pregunta	Este servicio está habilitado para el usuario, permite proponer una pregunta ya sea abierta, selección única respuesta, selección múltiple respuesta, verdadero-falso y pregunta con expresiones a un profesor.
Convertir $\text{\LaTeX}$	Mediante la librería de javascript MathJax permite modelar el texto en ecuaciones matemáticas.

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Visualizar pregunta	Permite visualizar las matemáticas que se esté digitando en la pregunta y en la solución.
Guardar pregunta	Guarda las preguntas y soluciones en base de datos en formato látex.

Tabla 4 Casos de uso: Visualización de preguntas del taller con látex por el estudiante en el aula virtual.

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Presentar Taller	Servicio en el cual el estudiante presentara el taller de la materia asignado por parte del profesor.
Visualizar pregunta	Se visualiza la pregunta en formato látex de tal forma que sea más legible para el estudiante.
Responder pregunta	Se da respuesta a la pregunta asignada en el taller.
Enviar pregunta	Se enviar la pregunta para que sea calificada de acuerdo a la solución ya estipulada.

Tabla 5 Casos de uso: Visualización de preguntas del examen con látex por el estudiante en el aula virtual.

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Presentar examen	Este servicio permite al estudiante presentar el examen de la materia asignado por el profesor.
Visualizar pregunta	Se visualiza la pregunta en formato látex de tal forma que sea más legible para el estudiante.
Responder pregunta	Se da respuesta a la pregunta asignada en el taller.
Enviar pregunta	Se envía la pregunta para que sea calificada de acuerdo a la solución ya estipulada.

### 5.3.3 Servicio para exportar las notas del aula virtual en formato csv

Tabla 6 Casos de uso: Exportar notas en formato csv del aula virtual

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Ver notas	Servicio para almacenar las notas de las diferentes actividades asignadas en el aula, como talleres, laboratorios, quices, exámenes, definitiva.
Seleccionar actividad	Se selecciona una sola actividad de la cual desee generar un archivo .csv con las notas de todos los estudiantes de esa actividad.
Digitar datos	Se digitan tres datos que son año, periodo académico y la nota parcial.
Generar csv	Se genera un archivo en formato csv con las notas de los estudiantes separados por “;” los nombres de los estudiantes, la materia, el código de la materia y se almacena en el servidor
Explorar archivos	Se listan los archivos de las actividades a la cual haya generado el archivo .csv
Descargar archivos	Permite descargar el archivo .csv seleccionado y almacenarlo en el computador
Eliminar archivo	Permite eliminar el archivo del servidor.

### 5.3.4 Servicio para la construcción y modificación del libro del aula virtual

Tabla 7 Casos de uso: Creación y modificación del libro virtual por parte del profesor

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Gestionar libros aula	Lista los libros que el profesor ha creado anteriormente.
Crear libro	Se selecciona o se digita el nombre del libro y se selecciona un grupo colaborador.
Modificar libro	Permite cambiar el nombre del libro y el grupo colaborador.
Eliminar libro	Permite eliminar todo lo que está incluido en el libro, es decir, unidades, temas y recursos (pdf, imágenes, videos, entre otros) de dicho libro.
Gestionar unidad	Se listan las unidades que pertenecen al libro seleccionado.
Agregar unidad	Se agrega el número de unidad, el nombre de la unidad, semana inicial y semana final.
Editar Unidad	Permite modificar el nombre del libro, las semanas inicial y final.
Eliminar unidad	Elimina todo lo que contenga una unidad, es decir, los temas y los recursos de cada tema de esa unidad.
Gestionar temas	Lista los temas de la unidad seleccionada.

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Gestionar archivos	Lista los archivos contenidos en el tema seleccionado. Permite la carga de diferentes archivos (pdf, imágenes, animaciones, videos, presentaciones) que apoyen el tema. Creación de carpetas, eliminar archivos, ver detalles, renombrar el archivo, mover archivos a otras carpetas que tenga creadas dentro del tema.
Gestionar enlaces	Lista los enlaces del tema seleccionado y permite subir enlaces que sirvan de apoyo al tema, digita el nombre o descripción y agrega la url de interés.
Editar tema	Permite cambiar el nombre del tema
Eliminar tema	Elimina los archivos que pertenecen al tema tanto del servidor como de bases de datos.

Tabla 8 Casos de uso: Creación y modificación del libro virtual por un grupo colaborador.

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Gestionar libros aula	Lista los libros al cual se le ha asignado a un grupo colaborador.
Modificar libro	Permite cambiar el nombre del libro y el grupo colaborador.
Eliminar libro	Permite eliminar todo lo que está incluido en el libro, es decir, unidades, temas y recursos (pdf, imágenes, videos, entre otros) de dicho libro.
Gestionar unidad	Se listan las unidades que pertenecen al libro seleccionado.
Agregar unidad	Se agrega el número de unidad, el nombre de la unidad, semana inicial y semana final.

TÍTULO	FUNCIONES PRIMARIAS
Editar Unidad	Permite modificar el nombre del libro, las semanas inicial y final.
Eliminar unidad	Elimina todo lo que contenga una unidad, es decir, los temas y los recursos de cada tema de esa unidad.
Gestionar temas	Lista los temas de la unidad seleccionada.
Gestionar archivos	Lista los archivos contenidos en el tema seleccionado. Permite la carga de diferentes archivos (pdf, imágenes, animaciones, videos, presentaciones) que apoyen el tema. Creación de carpetas, eliminar archivos, ver detalles, renombrar el archivo, mover archivos a otras carpetas que tenga creadas dentro del tema.
Gestionar enlaces	Lista los enlaces del tema seleccionado y permite subir enlaces que sirvan de apoyo al tema, digita el nombre o descripción y agrega la url de interés.
Editar tema	Permite cambiar el nombre del tema
Eliminar tema	Elimina los archivos que pertenecen al tema tanto del servidor como de bases de datos.

## 5.4 DISEÑO Y ANÁLISIS

### Diagramas Entidad/Relación de los servicios desarrollados

#### 5.4.1 Servicio para la construcción y modificación del libro del aula virtual

Figura 13 Diagrama E/R: Construcción y modificación del libro virtual



## 5.5 DESCRIPCIÓN DE LAS ENTIDADES

Descripción de las entidades creadas y modificadas en las bases de datos Diamante ubicada en el servidor de cada portal y Poseidón ubicada en el servidor del portal web de la Vicerrectoría Académica.

Tabla 9 Descripción de las entidades

ENTIDAD	DESCRIPCIÓN
TP_RecursosOva	Contiene la información de los recursos que pertenecen a un tema de un libro en específico.

## 5.6 MODELO DE PROCESOS DEL SISTEMA

Para una mejor interpretación de los modelos de los procesos del sistema se realizaron los diagramas de secuencia necesarios para cada caso de uso en los que se explica con detalle los pasos para el funcionamiento de cada uno de los servicios (Ver anexo A).

## 5.7 IMPLEMENTACIÓN, IMPLANTACIÓN Y PRUEBAS GENERALES

Para la implementación de los servicios se utilizaron las siguientes herramientas:

- Lenguaje de marcado para la elaboración de páginas web, HTML.
- Lenguaje de programación orientado a la web, JSP.
- Lenguaje de programación interpretado orientado a objetos, JavaScript.
- Hojas de estilo en cascada, CSS.
- Lenguaje Java.
- NetBeans, IDE para desarrollar las clases de Java y los archivos JSP.
- Servidor Apache Tomcat.
- Manejador de Base de datos, MySQL 5.7.

Haciendo uso de las anteriores herramientas, con la asesoría y seguimiento del director de proyecto se diseñaron los nuevos servicios y se estructuraron los datos para el primer prototipo. También se tomaron en cuenta sugerencias recibidas por miembros del Grupo Calumet, para realizar un posterior refinamiento a las interfaces.

Para el desarrollo se trabajó con la base de datos "Diamante" ubicada en todos los servidores de los portales de las escuelas, y con la base de datos centralizada "Poseidón" ubicada en el servidor del portal web de la Vicerrectoría Académica. En las cuales se crearon las tablas necesarias para el funcionamiento de los servicios. Se llevaron a cabo pruebas para cada subsistema propuesto, verificando que el resultado correspondiera con lo esperado, de esta forma, se evidenció el correcto funcionamiento en la captura de datos, selección de ítems y almacenamiento de información.

La implantación de los servicios se realizó en primer lugar en el portal web de pruebas del Grupo Calumet, donde se realizan las pruebas a todos los componentes. Finalmente se implementan en todos los portales web a los cuales el Grupo Calumet presta soporte.

## **5.8 MANTENIMIENTO Y ADMINISTRACIÓN**

**5.8.1 Actividades de Mantenimiento.** Dentro de las funciones que se realizan en la administración y mantenimiento de los servicios de los portales web se encuentra la tarea de corregir errores, las cuales se denominan incidencias, que se corrigen durante la primera fase como integrantes del Grupo Calumet.

**5.8.2 Actividades de Soporte a Usuarios.** Las escuelas cuentan en sus portales con un servicio de consultas y sugerencias, a través del cual se responden preguntas y se resuelven problemas de los usuarios. Diariamente se reciben consultas de estudiantes y profesores, que necesitan asesoría en el funcionamiento de servicios o soporte en el manejo de su usuario y contraseña. También se resuelven problemas de manera presencial, donde el usuario acude directamente a las oficinas del Grupo Calumet y es atendido directamente por alguno de sus miembros, que le prestan la asesoría necesaria. Algunas de las labores realizadas comúnmente son:

- Restablecimiento de contraseña a usuarios.
- Dar respuesta y solución a las inquietudes y problemas que tengan los usuarios de los portales respecto al uso de los servicios.
- Facilitar orientación a los estudiantes nuevos de las escuelas a las cuales presta sus servicios el grupo Calumet, en cuanto al registro en el portal y el uso de sus principales servicios.

**5.8.3 Actividades de Administración.** Dentro de las actividades realizadas por los miembros del Grupo Calumet, se encuentra la tarea de administrar los servidores de los portales de las escuelas, cada semestre se asignan nuevos administradores, los cuales cuentan con un usuario dentro de uno de los portales a los cuales presta soporte el grupo. Como administrador las tareas que se realizan comúnmente son:

- Realizar periódicamente copias de las bases de datos Diamante y División.
- Dar aval a las solicitudes de publicación de los usuarios en la cartelera para que puedan ser accedidas por la comunidad.
- Atender las sugerencias hechas por los usuarios del sistema a través del servicio de consultas y sugerencias.
- Actualizar periódicamente las bases de datos con respecto a la información que ofrece la División de Servicios de Información de la Universidad Industrial de Santander, para que el portal cuente con información actualizada.

## 6 PRUEBAS DEL SISTEMA

Para garantizar el correcto desarrollo de los nuevos servicios creados y a los que se le hizo reingeniería, se realizaron las siguientes pruebas:

### 6.1 PRUEBAS DE VERIFICACIÓN

Esta prueba es una de las más utilizadas en desarrollo de software mediante esta se aplican diferentes técnicas para detectar errores en el sistema antes de ser utilizado.

Se efectúa ejecutando paso a paso el proceso del servicio de manera que se explora cada funcionalidad que tiene el módulo desarrollado, realizando verificaciones de validación, los campos que son obligatorios no pueden quedar vacíos, por ejemplo. Las siguientes tablas describen las pruebas de cada caso de uso de los servicios que fueron desarrollados.

**6.1.1 Pruebas por componente.** Esta prueba se realizó para cada caso de uso de cada servicio desarrollado y descrito anteriormente:

- **Servicio para cancelar cualquier proyecto en desarrollo**

Tabla 10 Pruebas Realizadas: Cancelar proyecto de grado en desarrollo.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Buscar proyectos	Ingresar al servicio y buscar por nombre de estudiante, código estudiante, nombre director de proyecto y título del proyecto, para luego ver la lista de proyectos por cada criterio.	<input type="checkbox"/>

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Seleccionar proyecto	Acceder a la información del proyecto de grado.	<input type="checkbox"/>
Cancelar proyecto	Dar observación y cancelar el proyecto de grado y verificar que fue cancelado.	<input type="checkbox"/>

- **Servicios para implementar látex en el banco de preguntas del aula virtual**

Tabla 11 Pruebas Realizadas: Creación de pregunta por un profesor

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Convertir a látex	Digitar el texto en los tres formatos establecidos.	<input type="checkbox"/>
Visualizar pregunta	Verificar el texto digitado en el formato seleccionado.	<input type="checkbox"/>
Guardar pregunta	Visualizar en base de datos la pregunta almacenada con el formato digitado.	<input type="checkbox"/>

Tabla 12 Pruebas Realizadas: Creación de pregunta por un usuario

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Convertir a látex	Digitar el texto en los tres formatos establecidos.	<input type="checkbox"/>
Visualizar pregunta	Verificar el texto digitado en el formato seleccionado.	<input type="checkbox"/>
Guardar pregunta	Visualizar en base de datos la pregunta almacenada con el formato digitado.	<input type="checkbox"/>

Tabla 13 Pruebas Realizadas: Visualización de látex en el taller y examen

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Visualizar pregunta	Verificar las ecuaciones matemáticas de forma amigable para el usuario	<input type="checkbox"/>
Responder pregunta	Digitar la respuesta a la pregunta, para proceder con las demás preguntas ya con látex.	<input type="checkbox"/>

- **Servicio para exportar las notas del aula virtual en formato csv**

Tabla 14 Pruebas Realizadas: Exportar notas en formato csv del aula virtual

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Ver notas	Ingresar a ver las notas de cada actividad presentada en el aula.	<input type="checkbox"/>
Seleccionar actividad	Seleccionar una actividad (Taller, Examen, Laboratorios, Definitivas).	<input type="checkbox"/>
Digitar datos	Luego de seleccionar una actividad, se digitan los datos de año, periodo y nota parcial, para generar el archivo, verificando que los campos sean llenados.	<input type="checkbox"/>
Generar csv	Realiza la acción de pasar los datos digitados y las notas de los estudiantes a un archivo .csv	<input type="checkbox"/>
Explorar archivos	Se listan los archivos generados anteriormente en formato .csv	<input type="checkbox"/>
Descargar archivos	Al listar los archivos permite descargar el archivo que seleccione	<input type="checkbox"/>
Eliminar archivos	Se permite eliminar los archivos que no requiera y verificar que no se muestre.	<input type="checkbox"/>

- **Servicio para la construcción y modificación del libro del aula virtual**

Tabla 15 Pruebas Realizadas: Creación y modificación del libro virtual

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Gestionar libros aula	Se listan los libros que el profesor tiene ya creados	<input type="checkbox"/>
Crear libro	Ingresa el nombre del libro y selecciona un grupo colaborador (opcional).	<input type="checkbox"/>
Gestionar unidades	Se listan las unidades que pertenecen al libro seleccionado	<input type="checkbox"/>
Modificar libro	Se permite cambiar el nombre del libro y el grupo colaborador que desea.	<input type="checkbox"/>
Eliminar libro	Se permite eliminar todo el contenido del libro, desde unidades, temas y recursos del libro seleccionado y verificar que no se muestre.	<input type="checkbox"/>
Agregar Unidad	Se permite crear una nueva unidad para el libro seleccionado, digitando el número de la unidad, nombre, semana inicial y semana final estimada por el profesor.	<input type="checkbox"/>
Gestionar temas	Se listan los temas que pertenecen a la unidad seleccionada previamente.	<input type="checkbox"/>
Modificar unidad	Se permite editar el nombre de la unidad, la semana inicial y la semana final estimada por el profesor.	<input type="checkbox"/>

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Eliminar unidad	Se permite eliminar todo el contenido de la unidad seleccionada, desde los temas y los recursos de la unidad y verificar que no se muestre.	<input type="checkbox"/>
Agregar temas	Se permite agregar un tema a la unidad seleccionada, ingresando el número de tema y el nombre del tema.	<input type="checkbox"/>
Gestionar archivos	Se permite Subir archivos (pdf, Presentaciones, imágenes, videos, animaciones), crear carpetas dentro del tema seleccionado, eliminar archivos que no requiera, ver detalles de los archivos.	<input type="checkbox"/>
Gestionar enlaces	Se listan los enlaces ya creados y permite ingresar nuevos enlaces de interés, digitando el nombre del enlace y la url. También permite seleccionar los enlaces que ya no se requieran.	<input type="checkbox"/>
Editar tema	Se permite editar el nombre del tema seleccionado.	<input type="checkbox"/>
Eliminar tema	Se permite eliminar todos los recursos del tema seleccionado y el tema y verificar que no se muestre.	<input type="checkbox"/>

Tabla 16 Pruebas Realizadas: Creación y modificación del libro por un encargado

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Gestionar libros aula	Se listan los libros que el profesor tiene ya creados	<input type="checkbox"/>
Gestionar unidades	Se listan las unidades que pertenecen al libro seleccionado	<input type="checkbox"/>
Modificar libro	Se permite cambiar el nombre del libro y el grupo colaborador que desea.	<input type="checkbox"/>
Eliminar libro	Se permite eliminar todo el contenido del libro, desde unidades, temas y recursos del libro seleccionado y verificar que no se muestre.	<input type="checkbox"/>
Agregar Unidad	Se permite crear una nueva unidad para el libro seleccionado, digitando el número de la unidad, nombre, semana inicial y semana final estimada por el profesor.	<input type="checkbox"/>
Gestionar temas	Se listan los temas que pertenecen a la unidad seleccionada previamente.	<input type="checkbox"/>
Modificar unidad	Se permite editar el nombre de la unidad, la semana inicial y la semana final estimada por el profesor.	<input type="checkbox"/>
Eliminar unidad	Se permite eliminar todo el contenido de la unidad seleccionada, desde los temas y los recursos de la unidad y verificar que no se muestre.	<input type="checkbox"/>

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Agregar temas	Se permite agregar un tema a la unidad seleccionada, ingresando el número de tema y el nombre del tema.	<input type="checkbox"/>
Gestionar archivos	Se permite Subir archivos (pdf, Presentaciones, imágenes, videos, animaciones), crear carpetas dentro del tema seleccionado, eliminar archivos que no requiera, ver detalles de los archivos.	<input type="checkbox"/>
Gestionar enlaces	Se listan los enlaces ya creados y permite ingresar nuevos enlaces de interés, digitando el nombre del enlace y la url. También permite seleccionar los enlaces que ya no se requieran.	<input type="checkbox"/>
Editar tema	Se permite editar el nombre del tema seleccionado	<input type="checkbox"/>
Eliminar tema	Se permite eliminar todos los recursos del tema seleccionado y el tema y verificar que no se muestre.	<input type="checkbox"/>

## 7 CONCLUSIONES

1. La creación del servicio para cancelar cualquier proyecto de grado en desarrollo por parte del comité de trabajos de grado, le permite dar de baja aquellos proyectos que no muestren resultados satisfactorios o aquellos proyectos que no tengan un seguimiento constante por parte del estudiante.
2. La actualización a solicitud de las escuelas (Comité de trabajo de grado y Claustro de profesores), para ocultar el nombre del profesor evaluador del plan al autor, le otorga privacidad al evaluador en el momento de dar su opinión acerca del plan de proyecto de grado.
3. La implementación de un sistema de composición de texto Látex al banco de preguntas del aula, permitió generar estructuras matemáticas en las preguntas, respuestas y soluciones que facilita la visualización del texto con expresiones matemáticas tanto al profesor como a los estudiantes.
4. La mejora al servicio del aula virtual con la implementación de una nueva funcionalidad que permite generar en formato csv un archivo con las notas de las diferentes actividades que se realizan en el aula, facilita al docente subir las notas de los estudiantes al sistema de la universidad.
5. La implementación del control de sesión en el aula virtual, permite controlar el tiempo en la presentación de un taller, examen o un foro, para que avise al usuario en un momento dado el tiempo restante de la misma y así tomar la decisión de continuar o cerrar la sesión, de esta manera no afecta el desarrollo de las actividades mencionadas anteriormente.
6. La actualización de la creación y modificación del libro virtual, permite una interfaz más amigable, entendible y de fácil acceso al profesor o encargado.

## **8. RECOMENDACIONES**

Para mantener una buena comunicación con los usuarios de los portales, es fundamental para dar pronta y precisa respuesta a las sugerencias de los mismos dando un soporte oportuno a sus inquietudes.

Capacitar a los profesores sobre los cambios realizados en el servicio de trabajos de grado para que así puedan hacer uso de manera óptima de este servicio.

Capacitar a los profesores sobre el nuevo servicio de exportar las notas en formato csv y crear preguntas con látex para que puedan hacer uso de este mismo sin ningún problema.

Capacitar a los usuarios sobre las mejoras realizadas en el servicio de creación y modificación del libro virtual y el control de sesión en el aula, para que puedan hacer buen uso del mismo y ahorren tiempo al ejecutar tareas relacionadas con este.

## BIBLIOGRAFÍA

JAVASCRIPT-YA. Tutoriales del lenguaje de programación JavaScript. [En línea]. (Recuperado en 01 octubre 2017) Disponible en <http://www.tutorialesprogramacionya.com/javascriptya/index.php?inicio=0>

JOHNSON, James. Bases de datos: Modelos lenguajes y diseño. 1ª ed. Oxford, 2000. Presenta temas de teoría de bases de datos: modelos y métodos de acceso, administración, diseño de aplicaciones.

MANUALES. Manuales básicos de JSP. [En línea]. (Recuperado en 30 septiembre 2017). Disponible en <http://www.desarrolloweb.com/manuales/73/>

PROGRAMACIÓN. En este sitio se encuentran un completo curso para aprender java. [En línea]. (Recuperado en 01 octubre 2017). Disponible en [http://www.aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=68&Itemid=188](http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=68&Itemid=188)

PROGRAMACIÓN FÁCIL. Manual de programación para Java Jsp (Java Server Pages). [En línea]. (Recuperado en 01 octubre 2017). Disponible en [http://www.programacionfacil.com/java\\_servlets/start.html](http://www.programacionfacil.com/java_servlets/start.html)

STALLINGS, William. Sistemas Operativos: Aspectos Internos y Principios de Diseño. Madrid: Pearson Prentice Hall, 2005. Este libro se ocupa de los conceptos completos de las características de los sistemas operativos.

STUMPF, Robert, TEAGUE, Lavette. Object-Oriented Systems Analysis and Design with UML. Prentice Hall. 2004. Este libro introduce los conceptos y métodos del análisis y diseño de sistemas orientados a objetos.

SQL. En este sitio se encuentra una buena documentación con ejemplos de sql. [En línea]. (Recuperado en 07 octubre 2017). Disponible en <http://www.1keydata.com/es/sql/>

WEITZENFELD, Alfredo. Ingeniería de Software Orientada a Objetos con UML, JAVA e Internet. Thomson International, 2003. En este libro se encuentra información sobre desarrollo de software orientado a objetos.

W3SCHOOLS. Tutoriales de cómo programar en html de gran ayuda en detalles básicos y sencillos. [En línea]. (Recuperado en 07 octubre 2017). Disponible en <http://www.w3schools.com/html/>

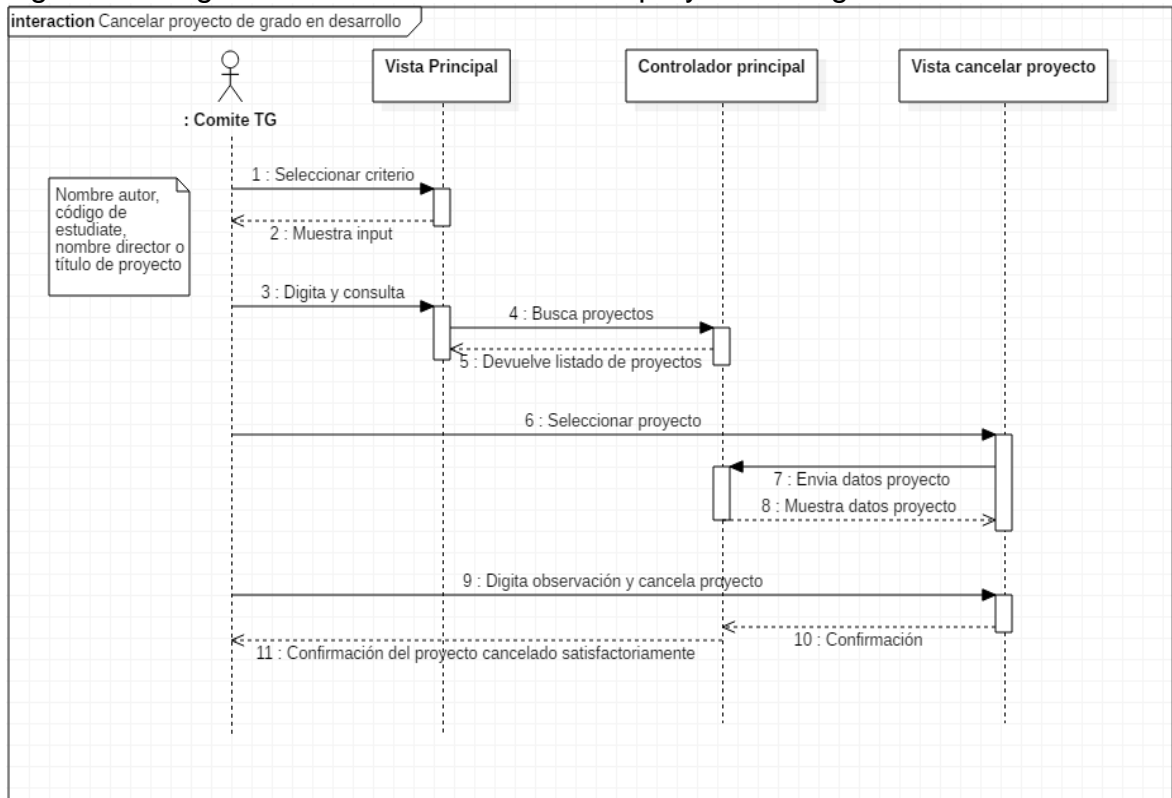
WIKISPACES. Información acerca de la Arquitectura Cliente Servidor. [En línea]. (Recuperado en 02 octubre 2017). Disponible en URL:<http://g701giadar.wikispaces.com/Arquitctura+Cliente+Servidor>

## ANEXOS

### Anexo A. Modelo de Procesos del Sistema

#### Servicio para la cancelación de proyectos en desarrollo

Figura 14 Diagrama de secuencia: Cancelar proyectos de grado



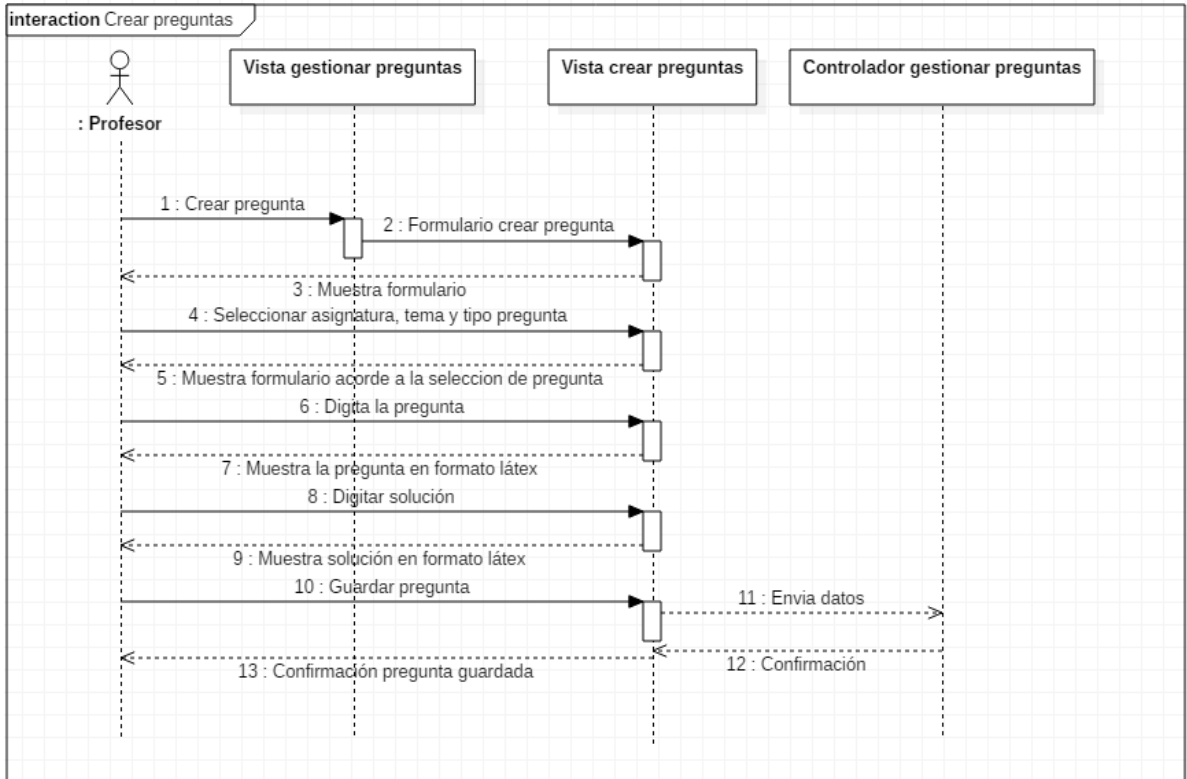
1. El comité de trabajos de grado selecciona un criterio para este caso nombre del autor, código de estudiante, nombre del director de proyecto o título del proyecto de grado.
2. La vista principal muestra un input para digitar el criterio seleccionado.
3. El comité de trabajos de trago digita el criterio y le da al botón buscar.
4. La vista principal le envía al controlador principal los datos digitados.
5. El controlador principal busca en bases de datos los proyectos que coincidan

con el criterio y devuelve la información.

6. El comité de trabajos de grado selecciona el proyecto a cancelar.
7. La vista cancelar proyecto le envía los datos al controlador principal para que busque la información del proyecto de grado seleccionado.
8. El controlador le manda los datos del proyecto seleccionado a la vista cancelar proyecto.
9. El comité de trabajos de grado ve la información del proyecto, da una observación y cancela el proyecto de grado.
10. La vista cancelar proyecto manda un mensaje de confirmación al controlador principal.
11. El controlador principal manda un mensaje al comité de trabajos de grado un mensaje de que cancelo el proyecto satisfactoriamente.

## Servicio de crear preguntas con $\text{\LaTeX}$ en el banco de preguntas del aula

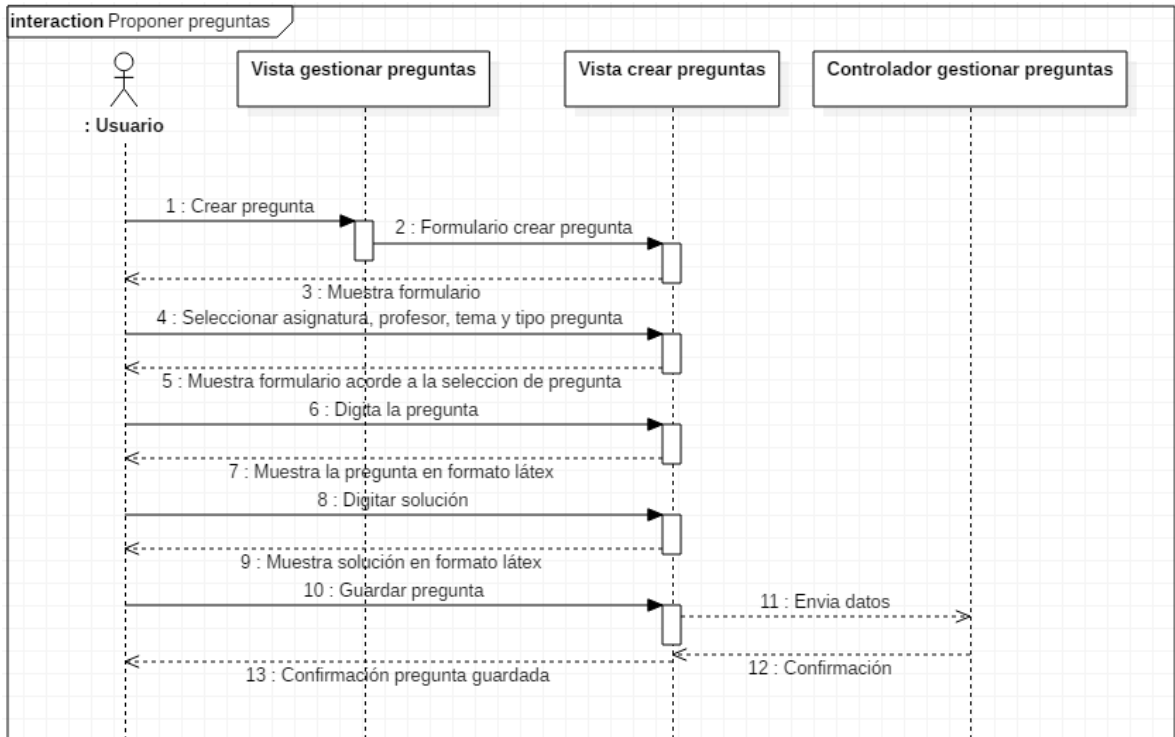
Figura 15 Diagrama de secuencia: Crear preguntas con  $\text{\LaTeX}$  por un profesor



1. El profesor selecciona crear preguntas.
2. La vista gestionar preguntas envía a la vista crear preguntas el formulario para crear una nueva pregunta.
3. La vista crear preguntas le muestra el formulario estándar al profesor donde le muestra tres select de asignatura, tema y tipo pregunta y un formulario para digitar las preguntas.
4. El profesor selecciona la asignatura, el tema y el tipo de pregunta que desea crear.
5. La vista crear preguntas le muestra al profesor un formulario de acuerdo al tipo de pregunta que escogió.
6. El profesor digita la pregunta con los tres formatos recomendados para digitar con  $\text{\LaTeX}$ .

7. El vista crear preguntas le muestra al profesor la pregunta en el formato  $\text{l\`atex}$  que digito.
8. El docente digita la soluci3n con los formatos recomendados para digitar texto con  $\text{l\`atex}$ .
9. La vista crear pregunta le muestra la soluci3n en el formato  $\text{l\`atex}$  que la digito.
10. El profesor guarda la pregunta.
11. La vista crear pregunta le envía los datos al controlador crear pregunta para que guarde en base de datos la pregunta que se acaba de crear.
12. El controlador crear pregunta le devuelve una confirmaci3n a la vista crear pregunta.
13. La vista crear pregunta le muestra una confirmaci3n de que la pregunta fue creada y guardada exitosamente.

Figura 16 Diagrama de secuencia: Crear preguntas con látex por un usuario

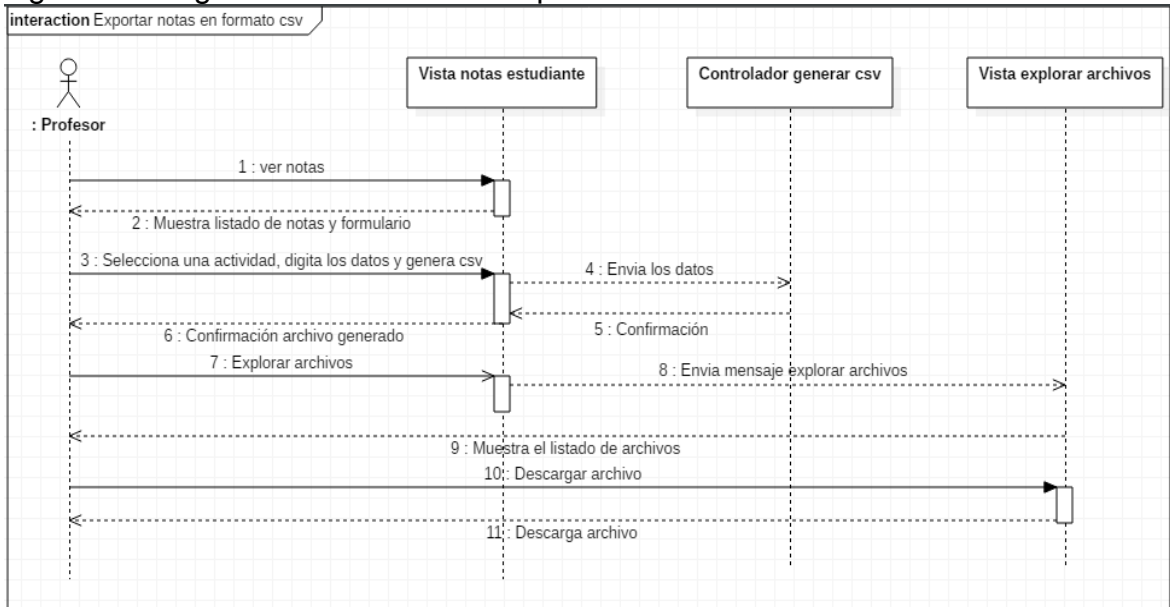


1. El usuario selecciona crear preguntas.
2. La vista gestionar preguntas envía a la vista crear preguntas el formulario para crear una nueva pregunta.
3. La vista crear preguntas le muestra el formulario estándar al usuario donde le muestra los select de asignatura, profesor, tema y tipo pregunta y un formulario para digitar las preguntas.
4. El usuario selecciona la asignatura, el profesor, el tema y el tipo de pregunta que desea crear.
5. La vista crear preguntas le muestra al usuario un formulario de acuerdo al tipo de pregunta que escogió.
6. El usuario digita la pregunta con los tres formatos recomendados para digitar con látex.
7. El vista crear preguntas le muestra al usuario la pregunta en el formato látex que digito.
8. El usuario digita la solución con los formatos recomendados para digitar texto con látex.

9. La vista crear pregunta le muestra la solución en el formato  $\text{l\`a t e x}$  que la digito.
10. El usuario guarda la pregunta.
11. La vista crear pregunta le envía los datos al controlador crear pregunta para que guarde en base de datos la pregunta que se acaba de crear.
12. El controlador crear pregunta le devuelve una confirmación a la vista crear pregunta.
13. La vista crear pregunta le muestra una confirmación de que la pregunta fue creada y guardada exitosamente.

## Servicio exportar notas en formato csv

Figura 17 Diagrama de secuencia: Exportar notas del aula

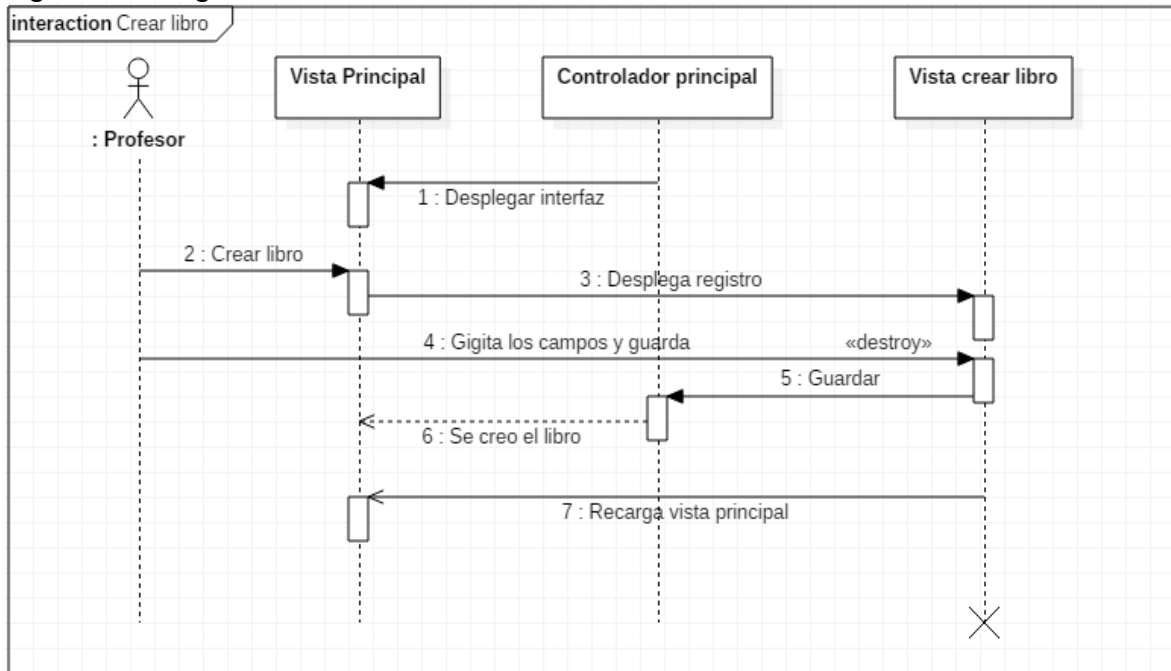


1. El profesor hace click al ver notas.
2. La vista notas estudiante le muestra al profesor el listado de las notas y un formulario de generar notas en formato csv para digitar.
3. El profesor selecciona una actividad que desea generar el archivo csv, luego digita la información solicitada y oprime el botón generar csv.
4. La vista notas estudiante envía los datos al controlador generar csv.
5. El controlador generar csv le manda a la vista notas estudiante una confirmación de los datos.
6. La vista notas estudiante devuelve un mensaje al profesor de que el archivo se generó con éxito.
7. El profesor da click en el botón explorar archivos.
8. La vista generar archivos le manda el mensaje de explorar archivos a la vista explorar archivos.
9. La vista explorar archivos le muestra al profesor una tabla con los archivos que ha generado.

10. El profesor selecciona descargar archivos.
11. La vista explorar archivos descarga el archivo al profesor.

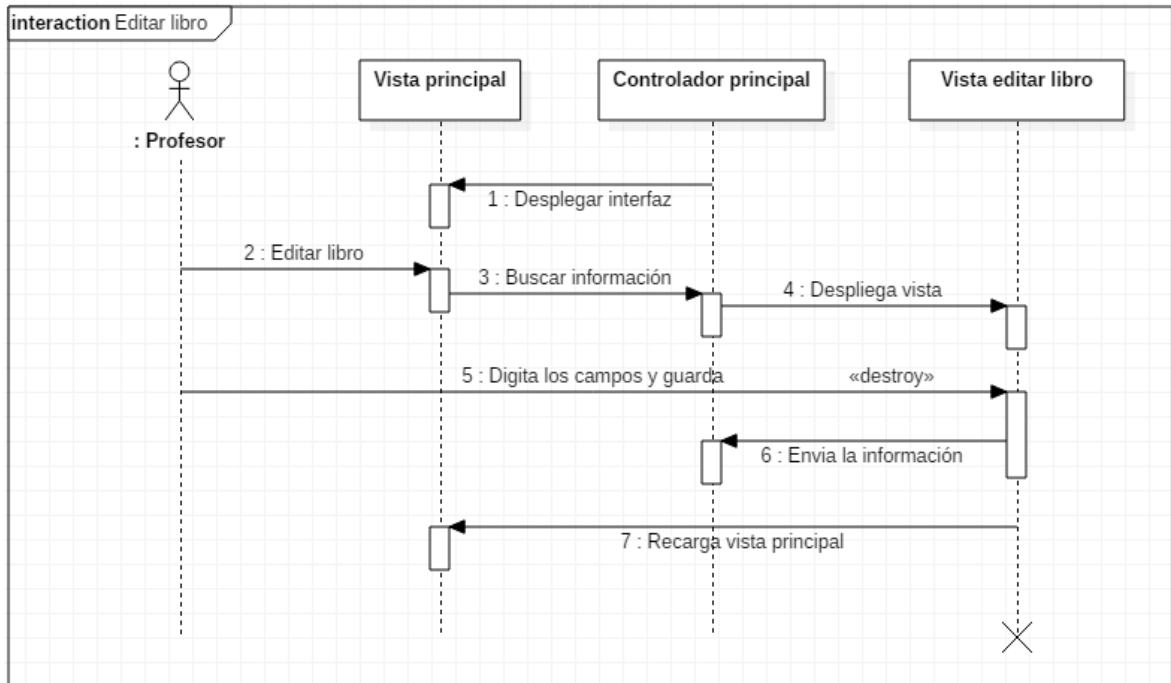
### Servicio creación y modificación del libro virtual

Figura 18 Diagrama de secuencia: Crear libro



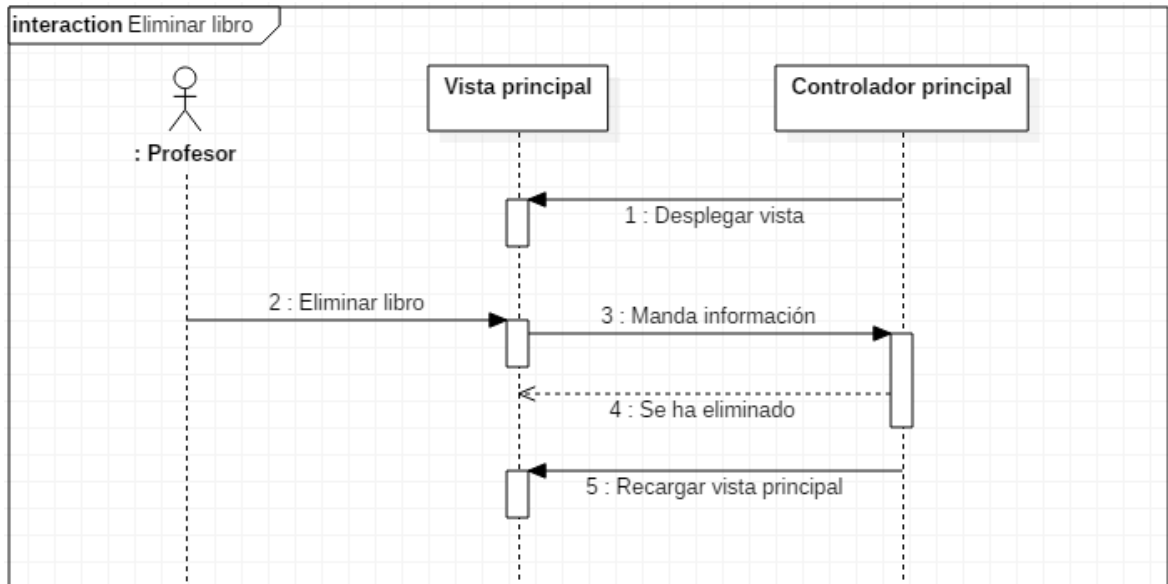
1. El controlador principal despliega la interfaz gráfica de la vista principal.
2. El profesor da clic en el botón crear nuevo libro en la vista principal
3. La vista principal despliega un registro en la vista de crear libro.
4. El profesor digita los campos de nombre y asigna o no grupo colaborador y envía el registro.
5. La vista Crear libro envía al controlador principal la información del libro a guardar
6. El controlador principal guarda en bases de datos la información y responde a la vista principal “Se ha creado el libro”.
7. Antes de cerrarse la vista crear libro actualiza la vista principal.

Figura 19 Diagrama de secuencia: Editar libro



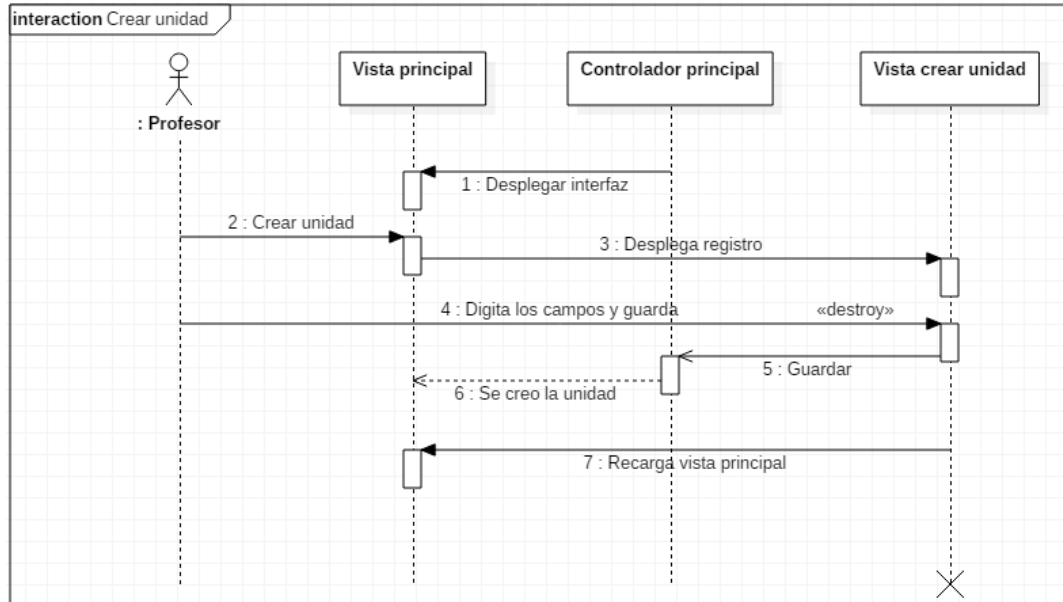
1. El controlador principal despliega la interfaz gráfica de la vista principal.
2. El profesor da clic en el botón editar libro en la vista principal
3. La vista principal busca la información del libro seleccionado al controlador principal.
4. El controlador principal despliega la vista con los datos a la vista editar libro.
5. El profesor modifica los campos pertinentes y envía el registro.
6. La vista editar libro envía al controlador principal la información del libro a modificar y el controlador principal guarda en bases de datos la información.
7. Antes de cerrarse la vista editar actualiza la vista principal.

Figura 20 Diagrama de secuencia: Eliminar libro



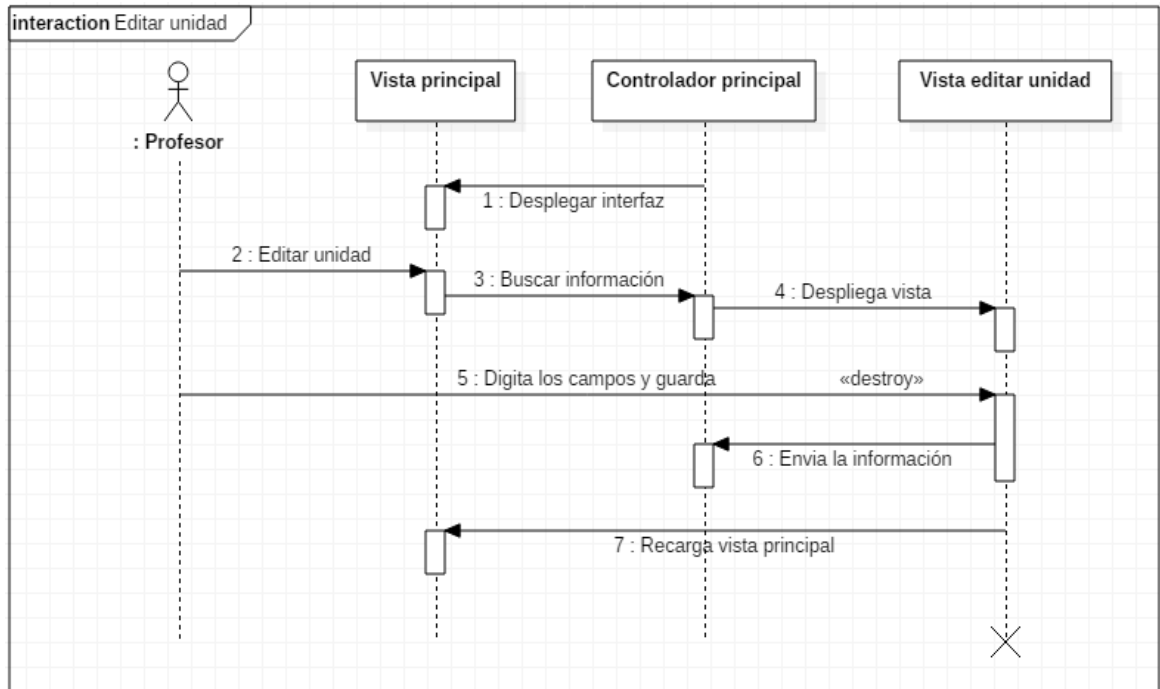
1. El controlador principal despliega la interfaz a la vista principal.
2. El profesor selecciona el botón eliminar libro en la vista principal.
3. La vista principal manda la información al controlador principal.
4. El controlador principal valida la información y manda una confirmación a la vista principal.
5. El controlador principal antes de mandar la confirmación recarga la vista principal.

Figura 21 Diagrama de secuencia: Crear unidad



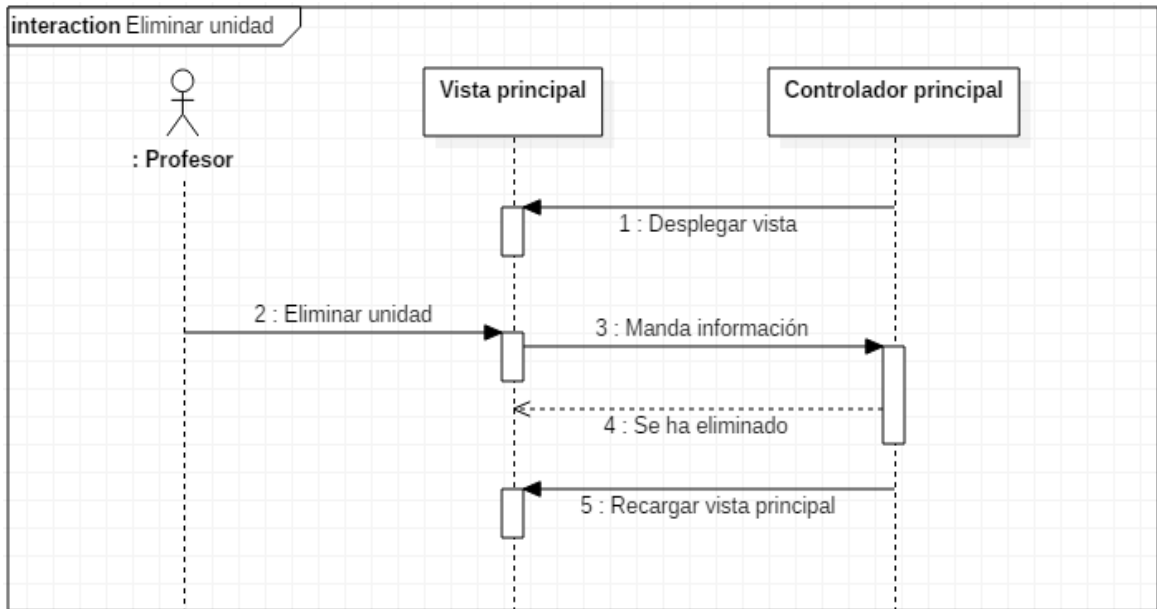
1. El controlador principal despliega la interfaz en la vista principal.
2. El profesor da clic en el botón agregar unidad en la vista principal
3. La vista principal despliega un registro en la vista de crear unidad.
4. El profesor digita los campos de número de unidad, nombre, semana inicial y final y envía el registro.
5. La vista Crear unidad envía al controlador principal la información de la unidad a guardar
6. El controlador principal guarda en bases de datos la información y responde a la vista principal "Se ha creado la unidad".
7. Antes de cerrarse la vista crear unidad actualiza la vista principal.

Figura 22 Diagrama de secuencia: Editar unidad



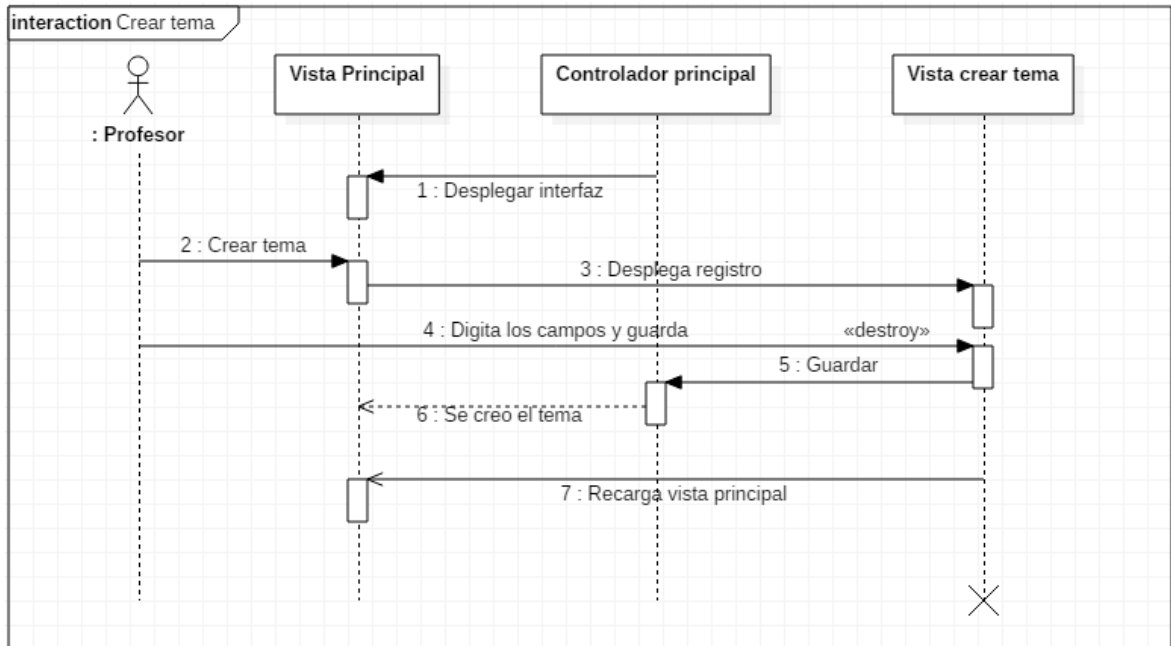
1. El controlador principal despliega la interfaz gráfica de la vista principal.
2. El profesor da clic en el botón editar unidad en la vista principal
3. La vista principal busca la información de la unidad seleccionada al controlador principal.
4. El controlador principal despliega la vista con los datos a la vista editar unidad.
5. El profesor modifica los campos pertinentes y envía el registro.
6. La vista editar unidad envía al controlador principal la información de la unidad a modificar y el controlador principal guarda en bases de datos la información.
7. Antes de cerrarse la vista editar actualiza la vista principal.

Figura 23 Diagrama de secuencia: Eliminar unidad



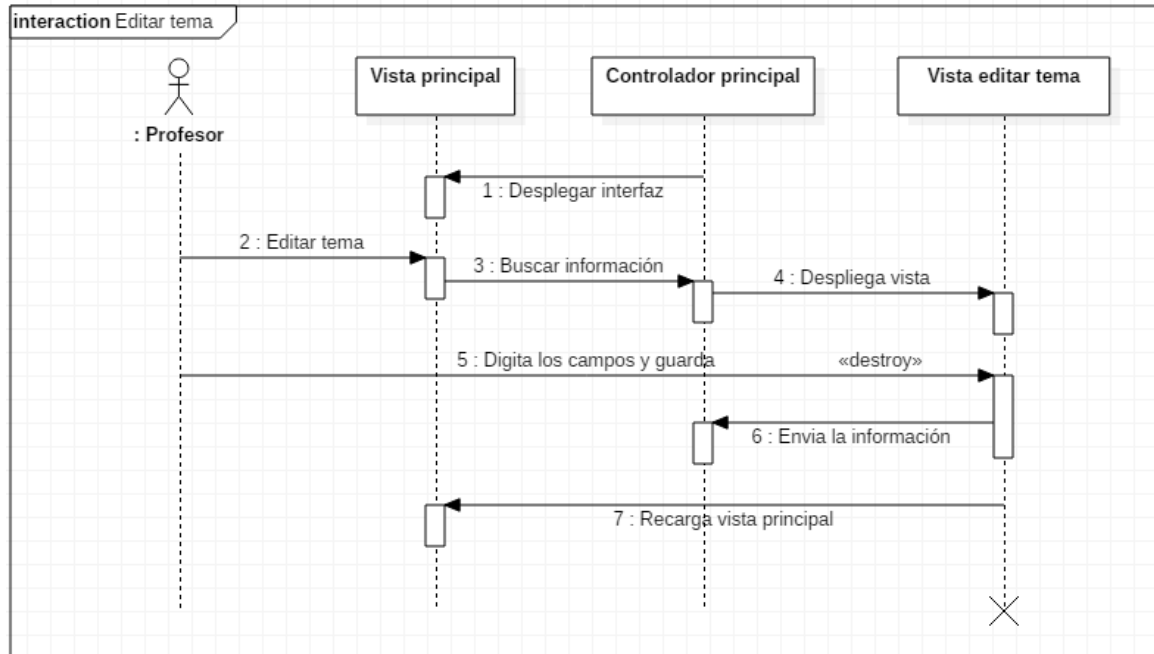
1. El controlador principal despliega la interfaz a la vista principal.
2. El profesor selecciona el botón eliminar unidad en la vista principal.
3. La vista principal manda la información al controlador principal.
4. El controlador principal valida la información y manda una confirmación a la vista principal.
5. El controlador principal antes de mandar la confirmación recarga la vista principal.

Figura 24 Diagrama de secuencia: Crear tema



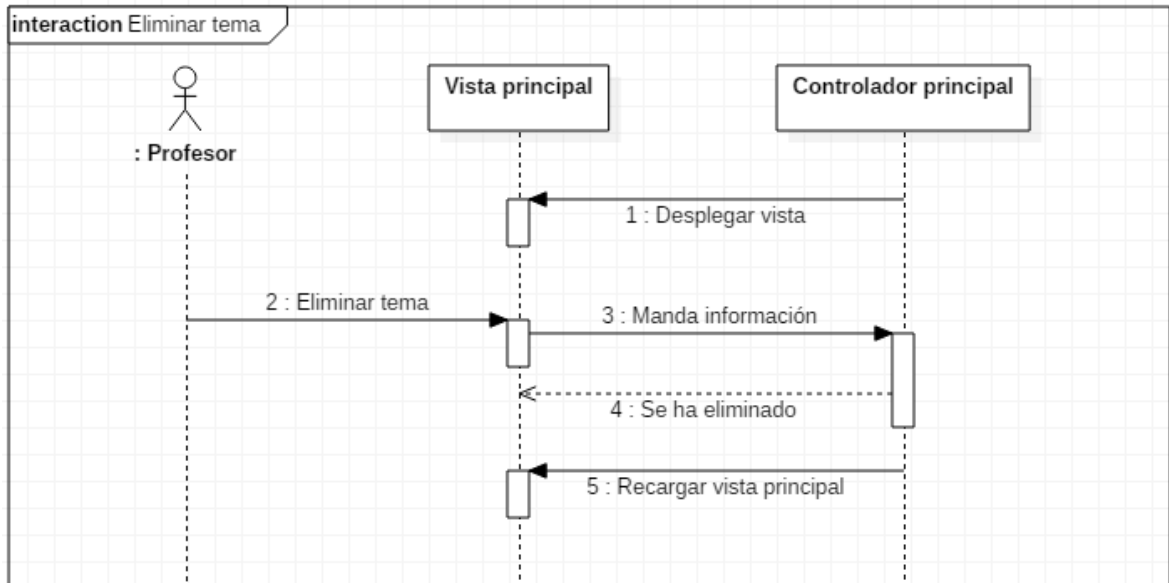
1. El controlador principal despliega la interfaz en la vista principal.
2. El profesor da clic en el botón agregar tema en la vista principal
3. La vista principal despliega un registro en la vista de crear tema.
4. El profesor digita los campos de número de tema, nombre y envía el registro.
5. La vista Crear tema envía al controlador principal la información del tema a guardar.
6. El controlador principal guarda en bases de datos la información y responde a la vista principal “Se ha creado el tema”.
7. Antes de cerrarse la vista crear tema actualiza la vista principal.

Figura 25 Diagrama de secuencia: Editar tema



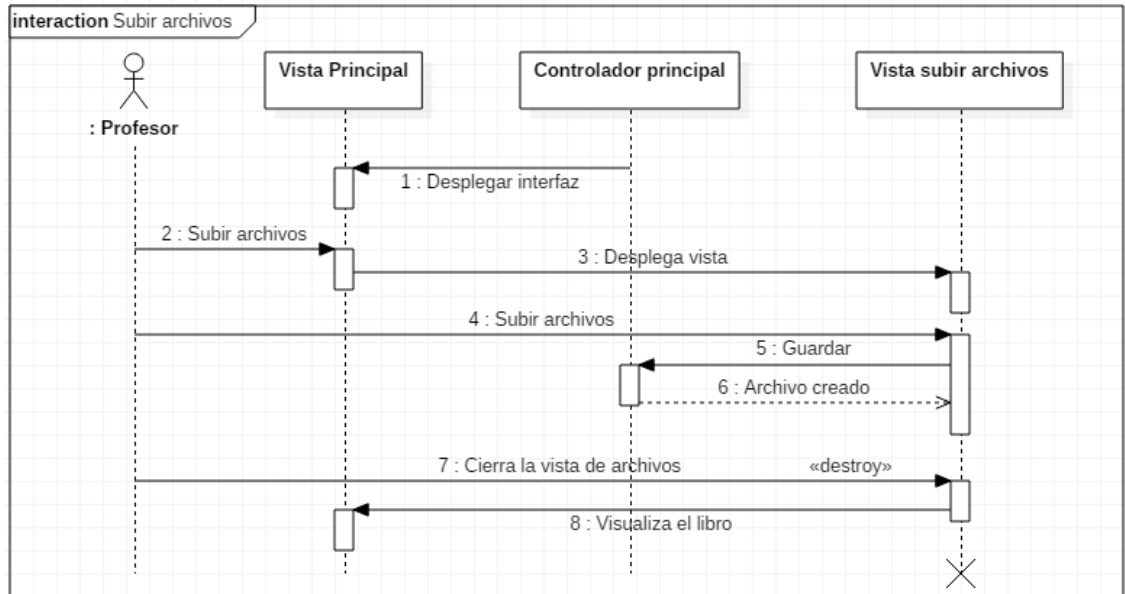
1. El controlador principal despliega la interfaz gráfica de la vista principal.
2. El profesor da clic en el botón editar tema en la vista principal
3. La vista principal busca la información del tema seleccionado al controlador principal.
4. El controlador principal despliega la vista con los datos a la vista editar tema.
5. El profesor modifica los campos pertinentes y envía el registro.
6. La vista editar tema envía al controlador principal la información del tema a modificar y el controlador principal guarda en bases de datos la información.
7. Antes de cerrarse la vista editar actualiza la vista principal.

Figura 26 Diagrama de secuencia: Eliminar tema



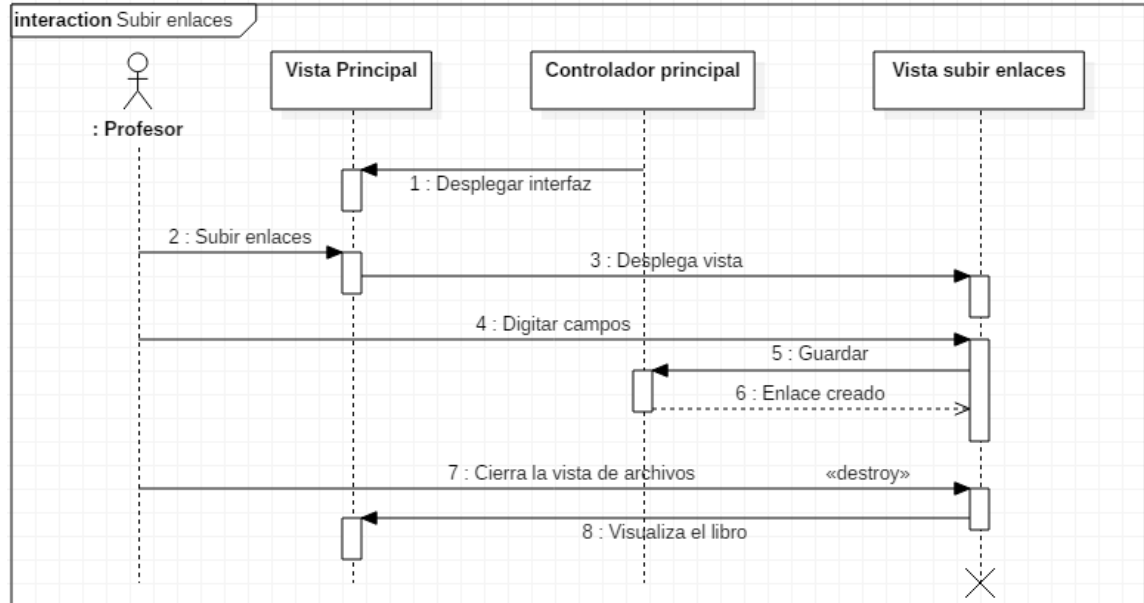
1. El controlador principal despliega la interfaz a la vista principal.
2. El profesor selecciona el botón eliminar tema en la vista principal.
3. La vista principal manda la información al controlador principal.
4. El controlador principal valida la información y manda una confirmación a la vista principal.
5. El controlador principal antes de mandar la confirmación recarga la vista principal.

Figura 27 Diagrama de secuencia: Subir archivos para un tema



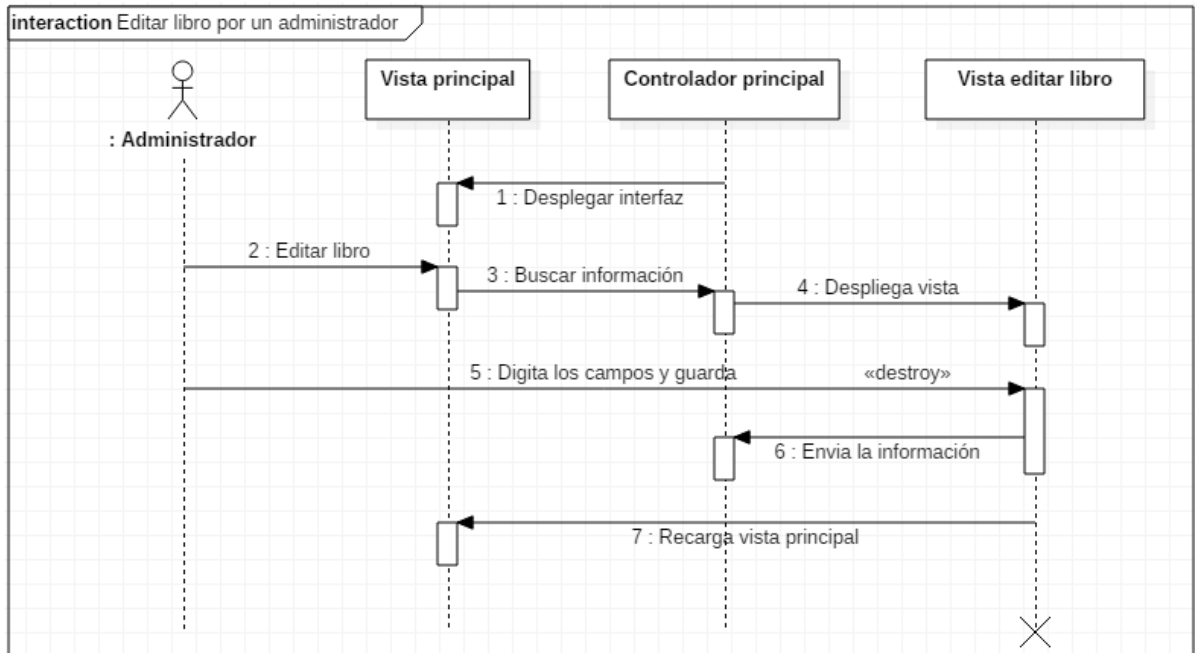
1. El controlador principal despliega la interfaz gráfica de la vista principal.
2. El profesor da click en subir archivos.
3. La vista principal despliega la vista en la vista subir archivos.
4. El profesor hace click en el botón subir archivos.
5. La vista subir archivos envía los datos al controlador principal.
6. El controlador principal guarda los datos del archivo en bases de datos y el servidor y devuelve archivo agregado con éxito.
7. El profesor cierra la vista de subir archivos.
8. La vista de subir archivos refresca la vista principal para mostrar el libro nuevamente.

Figura 28 Diagrama de secuencia: Subir enlaces para un tema



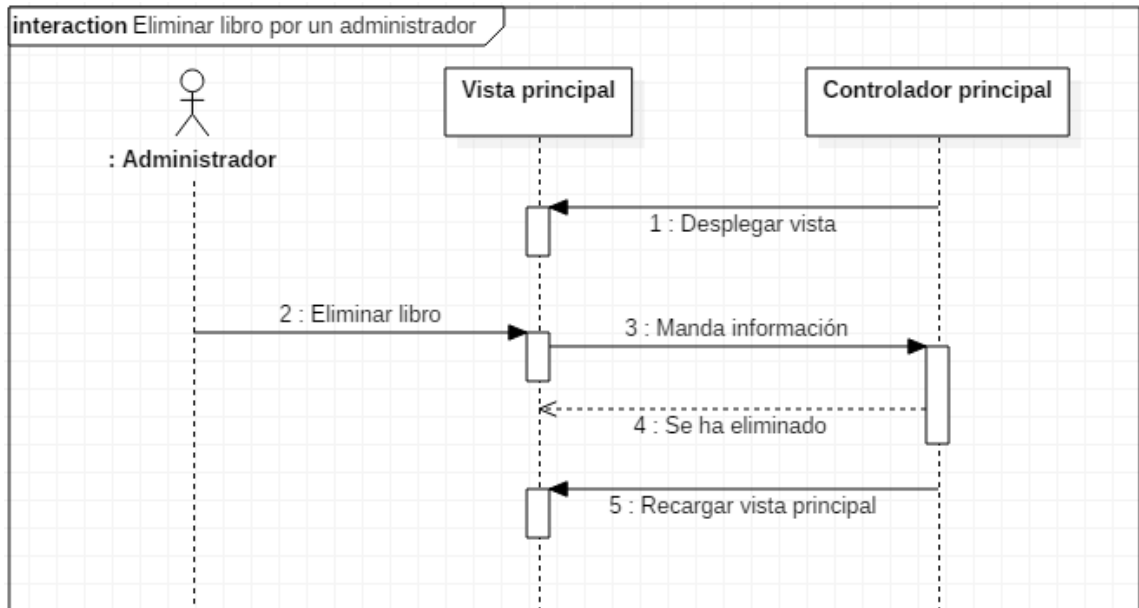
1. El controlador principal despliega la interfaz gráfica de la vista principal.
2. El profesor da click en subir enlaces.
3. La vista principal despliega la vista en la vista subir enlaces.
4. El profesor digita los campos del nombre o descripción y la url.
5. La vista subir enlaces envía los datos al controlador principal.
6. El controlador principal guarda los datos del nuevo enlace en bases de datos y devuelve el enlace ya creado a la vista subir enlaces.
7. El profesor cierra la vista subir enlaces.
8. La vista subir enlaces refresca la vista principal para mostrar el libro.

Figura 29 Diagrama de secuencia: Editar libro por un administrador



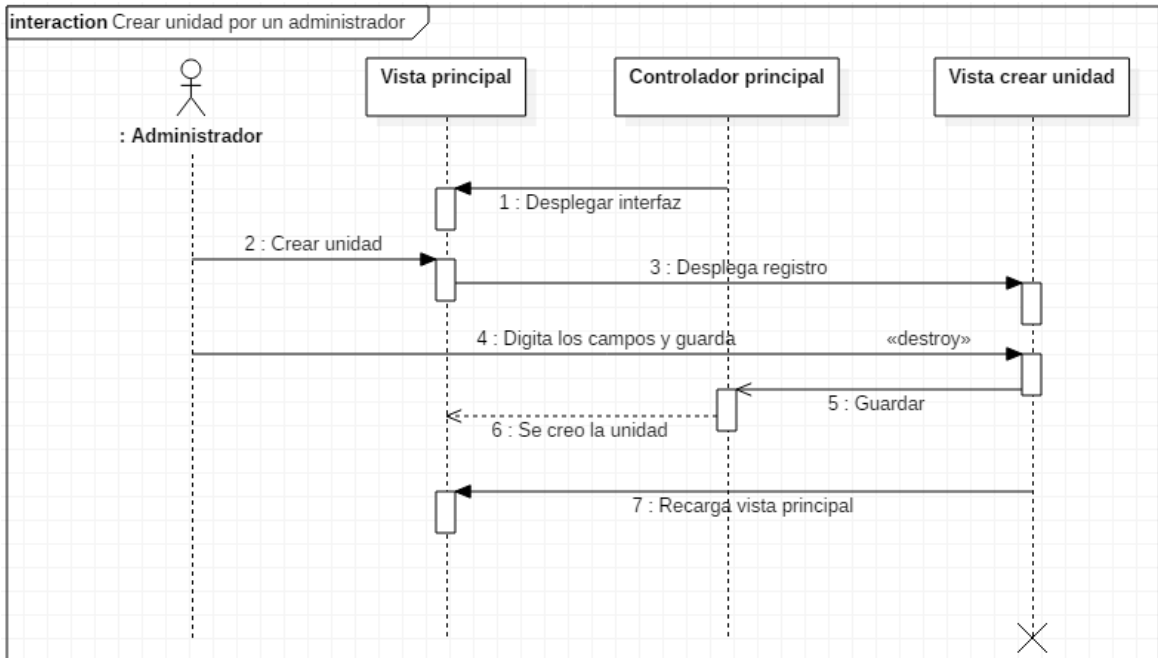
1. El controlador principal despliega la interfaz gráfica de la vista principal.
2. El administrador da clic en el botón editar libro en la vista principal
3. La vista principal busca la información del libro seleccionado al controlador principal.
4. El controlador principal despliega la vista con los datos a la vista editar libro.
5. El administrador modifica los campos pertinentes y envía el registro.
6. La vista editar libro envía al controlador principal la información del libro a modificar y el controlador principal guarda en bases de datos la información.
7. Antes de cerrarse la vista editar actualiza la vista principal.

Figura 30 Diagrama de secuencia: Eliminar libro por un administrador



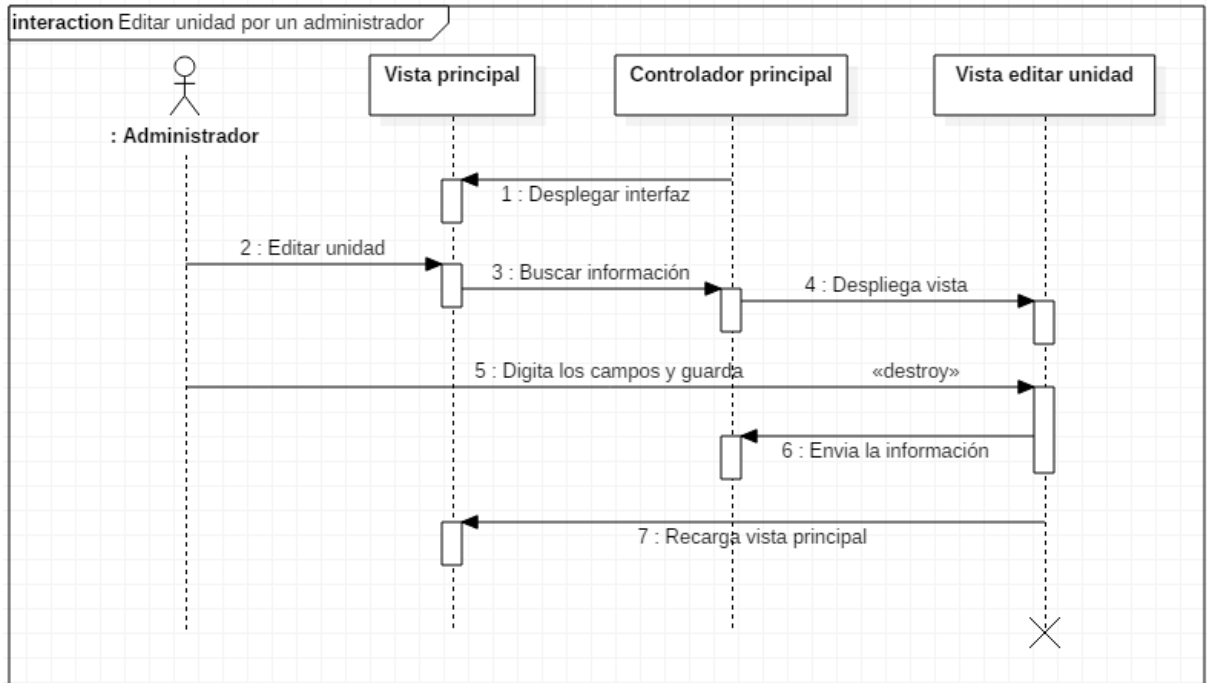
1. El controlador principal despliega la interfaz a la vista principal.
2. El administrador selecciona el botón eliminar libro en la vista principal.
3. La vista principal manda la información al controlador principal.
4. El controlador principal valida la información y manda una confirmación a la vista principal.
5. El controlador principal antes de mandar la confirmación recarga la vista principal.

Figura 31 Diagrama de secuencia: Crear unidad por un administrador



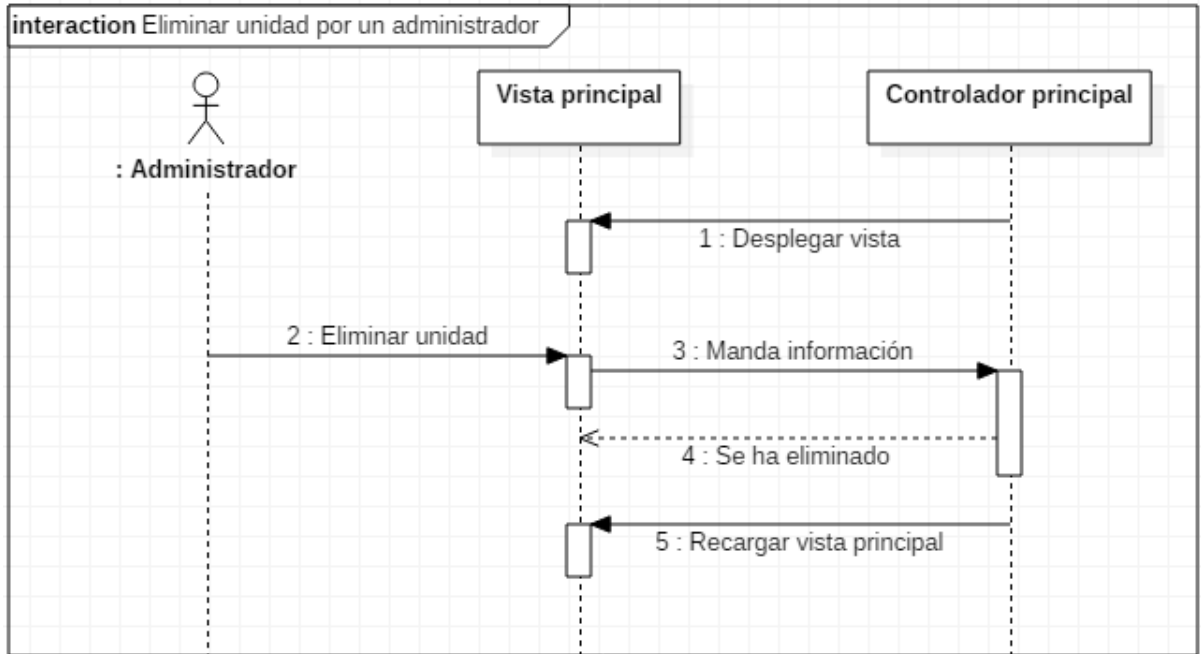
1. El controlador principal despliega la interfaz en la vista principal.
2. El administrador da clic en el botón agregar unidad en la vista principal
3. La vista principal despliega un registro en la vista de crear unidad.
4. El administrador digita los campos de número de unidad, nombre, semana inicial y final y envía el registro.
5. La vista Crear unidad envía al controlador principal la información de la unidad a guardar
6. El controlador principal guarda en bases de datos la información y responde a la vista principal "Se ha creado la unidad".
7. Antes de cerrarse la vista crear unidad actualiza la vista principal.

Figura 32 Diagrama de secuencia: Editar unidad por un administrador



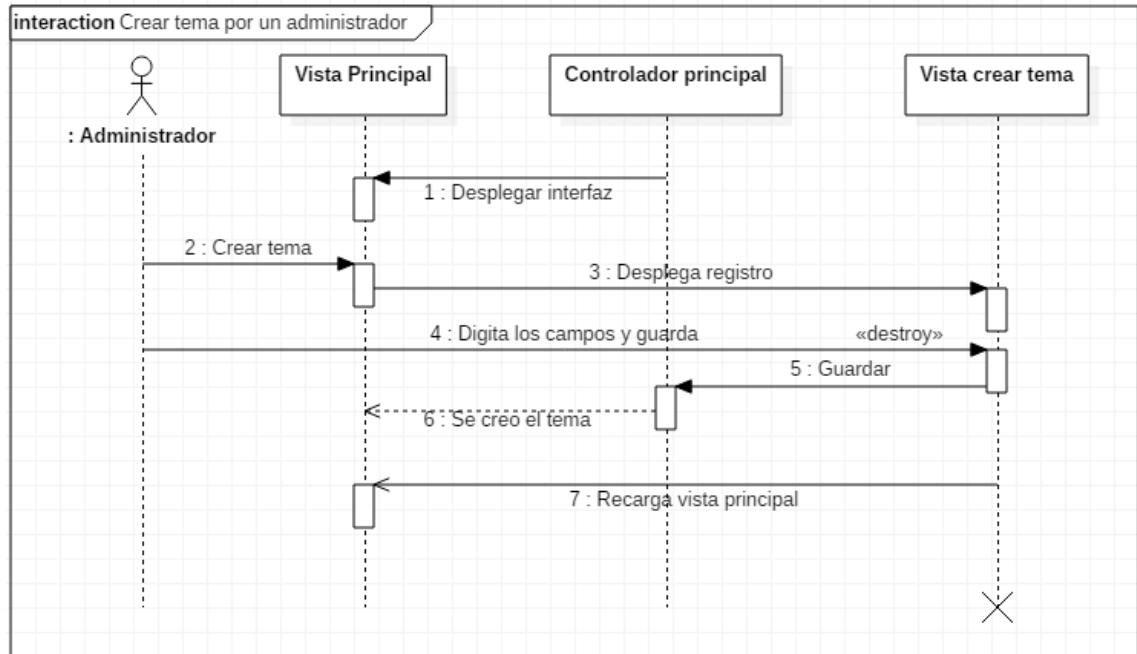
1. El controlador principal despliega la interfaz gráfica de la vista principal.
2. El administrador da clic en el botón editar unidad en la vista principal
3. La vista principal busca la información de la unidad seleccionada al controlador principal.
4. El controlador principal despliega la vista con los datos a la vista editar unidad.
5. El administrador modifica los campos pertinentes y envía el registro.
6. La vista editar unidad envía al controlador principal la información de la unidad a modificar y el controlador principal guarda en bases de datos la información.
7. Antes de cerrarse la vista editar actualiza la vista principal.

Figura 33 Diagrama de secuencia: Eliminar unidad por un administrador



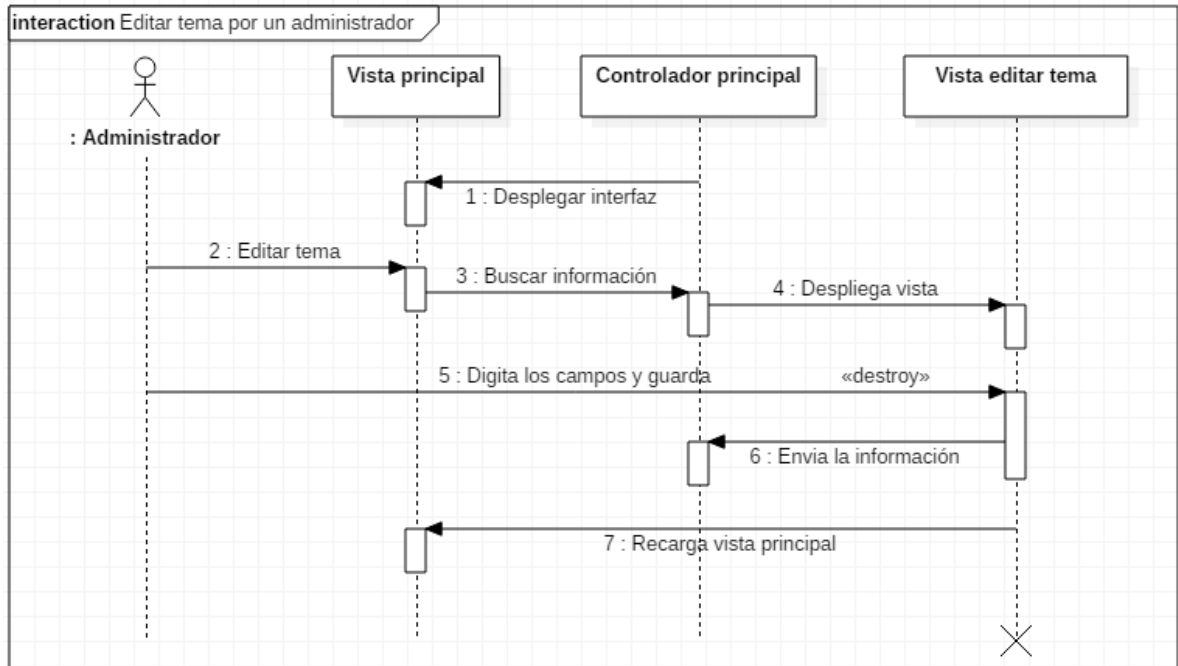
1. El controlador principal despliega la interfaz a la vista principal.
2. El administrador selecciona el botón eliminar unidad en la vista principal.
3. La vista principal manda la información al controlador principal.
4. El controlador principal valida la información y manda una confirmación a la vista principal.
5. El controlador principal antes de mandar la confirmación recarga la vista principal.

Figura 34 Diagrama de secuencia: Crear tema por un administrador



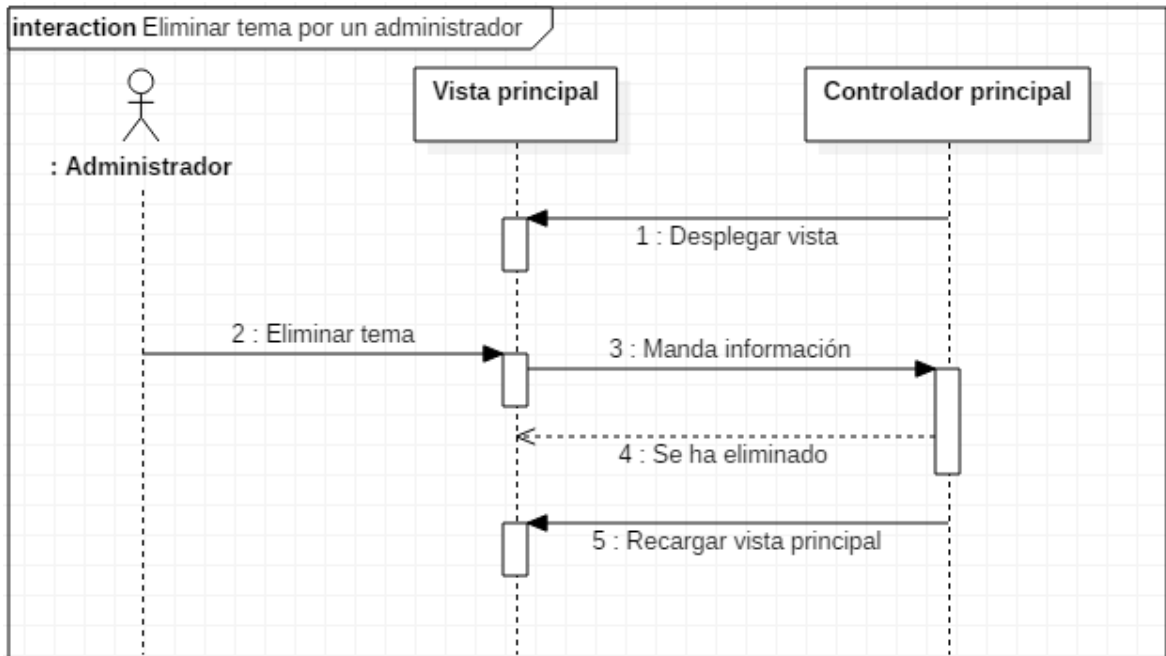
1. El controlador principal despliega la interfaz en la vista principal.
2. El administrador da clic en el botón agregar tema en la vista principal
3. La vista principal despliega un registro en la vista de crear tema.
4. El administrador digita los campos de número de tema, nombre y envía el registro.
5. La vista Crear tema envía al controlador principal la información del tema a guardar.
6. El controlador principal guarda en bases de datos la información y responde a la vista principal "Se ha creado el tema".
7. Antes de cerrarse la vista crear tema actualiza la vista principal.

Figura 35 Diagrama de secuencia: Editar tema por un administrador



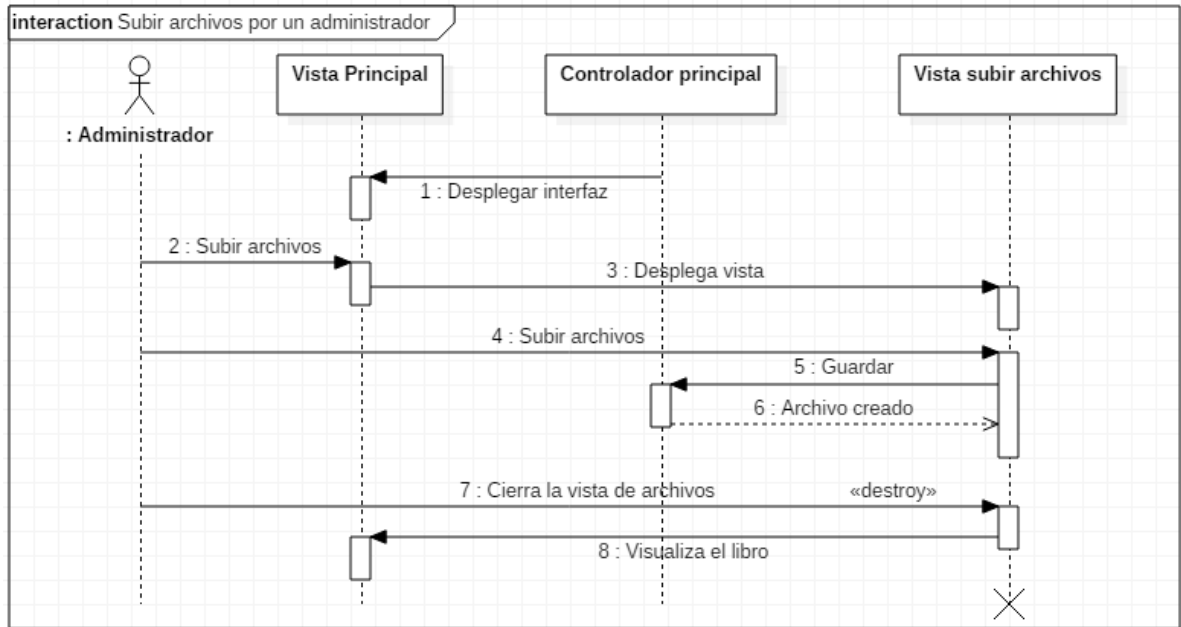
1. El controlador principal despliega la interfaz gráfica de la vista principal.
2. El administrador da clic en el botón editar tema en la vista principal
3. La vista principal busca la información del tema seleccionado al controlador principal.
4. El controlador principal despliega la vista con los datos a la vista editar tema.
5. El administrador modifica los campos pertinentes y envía el registro.
6. La vista editar tema envía al controlador principal la información del tema a modificar y el controlador principal guarda en bases de datos la información.
7. Antes de cerrarse la vista editar actualiza la vista principal.

Figura 36 Diagrama de secuencia: Eliminar tema por un administrador



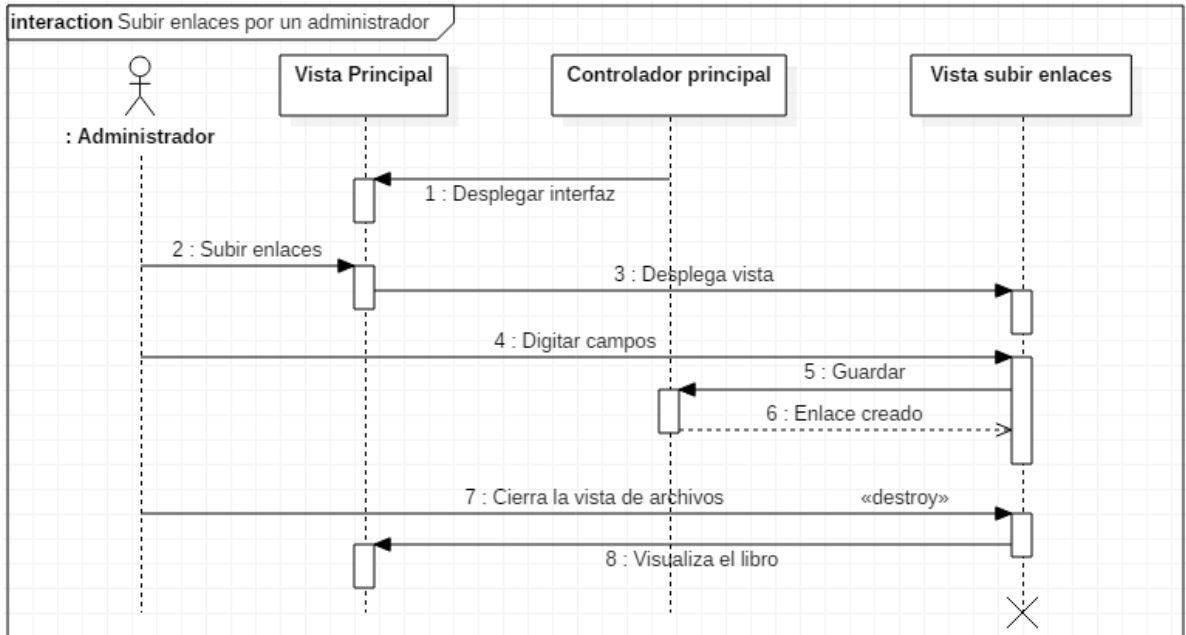
1. El controlador principal despliega la interfaz a la vista principal.
2. El administrador selecciona el botón eliminar tema en la vista principal.
3. La vista principal manda la información al controlador principal.
4. El controlador principal valida la información y manda una confirmación a la vista principal.
5. El controlador principal antes de mandar la confirmación recarga la vista principal.

Figura 37 Diagrama de secuencia: Subir archivos para un tema por un administrador



1. El controlador principal despliega la interfaz gráfica de la vista principal.
2. El administrador da click en subir archivos.
3. La vista principal despliega la vista en la vista subir archivos.
4. El administrador hace click en el botón subir archivos.
5. La vista subir archivos envía los datos al controlador principal.
6. El controlador principal guarda los datos del archivo en bases de datos y el servidor y devuelve archivo agregado con éxito.
7. El administrador cierra la vista de subir archivos.
8. La vista de subir archivos refresca la vista principal para mostrar el libro nuevamente.

Figura 38 Diagrama de secuencia: Subir enlaces para un tema por un administrador



9. El controlador principal despliega la interfaz gráfica de la vista principal.
10. El administrador da click en subir enlaces.
11. La vista principal despliega la vista en la vista subir enlaces.
12. El administrador digita los campos del nombre o descripción y la url.
13. La vista subir enlaces envía los datos al controlador principal.
14. El controlador principal guarda los datos del nuevo enlace en bases de datos y devuelve el enlace ya creado a la vista subir enlaces.
15. El administrador cierra la vista subir enlaces.
16. La vista subir enlaces refresca la vista principal para mostrar el libro.