

Documento de Propuesta de Diseño de Software I, II y III



Nombre de la propuesta: Creación de Contenidos.

Autores: Almanza Ávila José Daniel, Galeano Santamaría Fabio, Manjarrez Hernández, José Gabriel, Luna Genes Dor María, Valdelamar Vega Jesús David.

Colaboradores: Oscar Madera Bolaño , Eliana Romero Cuitiva, Katherine López Sierra y Dayana Cogollo Barba

Descripción del software

Este documento define el diseño de un componente de contenido para un software educativo, específicamente orientado hacia la creación de presentaciones. Este componente de contenido estará diseñado para mostrar la organización y visualización del material educativo en formato de presentaciones, permitiendo a los usuarios acceder de manera intuitiva a dichos recursos. El componente permitirá visualizar las presentaciones de manera organizada y jerarquizada, siguiendo los principios de accesibilidad, usabilidad y adaptabilidad. Este documento describe las características y funciones clave que tendrá este componente.

El módulo de contenidos sirve como el núcleo central de la plataforma, proporcionando acceso organizado y estructurado a las presentaciones a compartir por el docente. Este componente es fundamental para la experiencia del usuario, ya que concentra y presenta los recursos educativos de manera intuitiva. El módulo de contenidos facilita la entrega de información educativa de manera efectiva, permitiendo a los usuarios explorar, acceder, comentar y comprender las presentaciones de manera coherente y sistemática.

Este módulo educativo actúa como el punto central para la entrega y exploración de los materiales de aprendizaje en formato de presentaciones, proporcionando una plataforma eficiente y organizada para el proceso educativo.

Etapa 1: Diseño de la Aplicación y Análisis de Requisitos.

1. INTRODUCCIÓN.	5
• Propósito del Documento.	5
• Alcance del Proyecto.	5
• Definiciones y Acrónimos	5
2. DESCRIPCIÓN GENERAL	7
• Objetivos del Sistema.	7
• Funcionalidad general.	7
• Usuarios del Sistema.	7
• Restricciones.	9
3. REQUISITOS FUNCIONALES.	9
• Casos de Uso.	9
• Descripción detallada de cada caso de uso.	9
• Diagramas de Flujo de Casos de Uso	10
• Prioridad de Requisitos	22
4. REQUISITOS NO FUNCIONALES.	24
• Requisitos de Desempeño.	24
• Requisitos de Seguridad.	24
• Requisitos de Usabilidad.	25
• Requisitos de Escalabilidad.	25
5. MODELADO E/R.	26
• Diagrama de Entidad-Relación.	26
• Diagrama relacional.	26
• Script de modelo relacional.	26
• Descripción de entidades y relaciones.	26
• Reglas de integridad referencial.	26
• Colecciones (NoSQL).	26
6. ANEXOS.	26
• diagramas adicionales	26
• referencias	26

Etapa 2: Persistencia de datos con Backend.

7. INTRODUCCIÓN.	27
------------------	----

• Propósito de la etapa.	27
• Alcance de la etapa.	27
• Definiciones y acrónimos.	27
8. DISEÑO DE LA ARQUITECTURA DE BACKEND.	27
• Propósito de la etapa.	27
• Descripción de la arquitectura propuesta.	27
• Componentes del backend.	27
9. ELECCIÓN DE LA BASE DE DATOS.	27
10. IMPLEMENTACIÓN DEL BACKEND.	27
11. CONEXIÓN A LA BASE DE DATOS.	27
12. PRUEBAS DEL BACKEND	27

Etapa 3: Consumo de datos y desarrollo frontend.

13. INTRODUCCIÓN	28
14. CREACIÓN DE LA INTERFAZ DE USUARIO (UI)	28
15. PROGRAMACIÓN FRONTEND CON JAVASCRIPT (JS)	28
16. CONSUMO DE DATOS DESDE EL BACKEND.	28
17. INTERACCIÓN USUARIO-INTERFAZ.	28
18. PRUEBAS Y DEPURACIÓN DEL FRONTEND.	28
19. IMPLEMENTACIÓN DE LA LÓGICA DE NEGOCIO EN EL FRONTEND.	28
20. INTEGRACIÓN CON EL BACKEND.	28

Etapas 1: Diseño de la Aplicación y Análisis de Requisitos.

1. INTRODUCCIÓN.

- **Propósito del Documento.**

Este documento define el diseño de un componente de contenido para un software educativo. Este componente de contenido estará diseñado para mostrar la organización y visualización del material educativo que se presenta a los usuarios. El componente permitirá visualizar el contenido de manera organizada y jerarquizada, siguiendo los principios de accesibilidad, usabilidad y adaptabilidad. Este componente les permitirá a los usuarios navegar fácilmente por el contenido, el cual puede ser en formato de texto, multimedia o una combinación de ambos. Este documento describe las características y funciones clave que tendrá este componente.

- **Alcance del Proyecto.**

Este documento establece el alcance del proyecto para el desarrollo de un software educativo innovador, diseñado para transformar y enriquecer el proceso de aprendizaje. Este proyecto tiene como objetivo principal proporcionar una estructura de contenido de manera organizada, jerarquizada y fácil de manejar tanto como para docentes como para estudiantes.

Este software contará con un carrusel de imágenes, las cuales se irán moviendo al pasar el click por ellas y estará un menú desplegable conformado por imágenes, las cuales al pasar el click del mouse por la imagen, se desplegará un cuadro de información debajo de la imagen seleccionada.

A continuación, se presentan unas funciones que pueden ser integradas a futuro en la presente versión.

- Importar contenidos por parte de los estudiantes a la página (pdf, videos, presentaciones, documentos, infografías, etc.).

- **Definiciones y Acrónimos**

1. Componente de Contenido Educativo (CCE): Esta definición y este acrónimo se refiere al elemento diseñado dentro de un software educativo que tiene como objetivo principal organizar, presentar y permitir la visualización de materiales educativos de manera estructurada y jerarquizada. Este componente se adhiere a los principios de accesibilidad, usabilidad y adaptabilidad para mejorar la experiencia del usuario y facilitar la navegación a través de contenidos en formato de texto, multimedia o una combinación de ambos.

A través de contenidos en formato de texto, multimedia o una combinación de ambos.

2. DCCCSE (Documento de Definición del Componente de Contenido en Software Educativo): Esta definición y este acrónimo representa el documento que describe y establece los lineamientos para el diseño y funcionamiento del Componente de Contenido Educativo (CCE) en un software educativo. Este documento detalla las características y funciones clave que se implementarán en el CCE para optimizar la experiencia de aprendizaje.

3. Organización Jerarquizada Accesible (OJA): Esta definición y este acrónimo se refieren al enfoque de diseño del CCE que asegura que el material educativo esté estructurado de manera lógica y fácil de entender. Además, se garantiza que los usuarios puedan acceder y navegar por el contenido de manera efectiva, siguiendo los principios de accesibilidad.

4. Navegación Multiformato (NMF): Esta definición y este acrónimo describen la capacidad del Componente de Contenido Educativo (CCE) para permitir a los usuarios desplazarse y explorar contenido en varios formatos, incluyendo texto y multimedia, de manera intuitiva y sencilla.

5. Adaptabilidad en Contenido Educativo (ACE): Esta definición y este acrónimo se refieren a la característica del CCE que asegura que el material educativo se ajusta a las necesidades y preferencias del usuario, proporcionando una experiencia de aprendizaje personalizada.

2. DESCRIPCIÓN GENERAL

- **Objetivos del Sistema.**

Simplificar los procesos a los usuarios y visitantes, obteniendo un entorno accesible y flexible.

- **Funcionalidad general.**

Creación y edición de contenidos: El usuario puede crear contenidos, subir, editar y eliminarlo. El docente, en este caso al subir los contenidos todos sus estudiantes e invitados podrán visualizarlo.

Compartir contenidos: Permite compartir los contenidos educativos con otros usuarios, ya sea a través de enlaces o invitaciones, facilitando la colaboración entre estudiantes, docentes y otros profesionales de la educación

Comentar contenidos: Los usuarios podrán comentar los contenidos anteriormente cargados, esta funcionalidad permitirá tener un diálogo entre usuarios, sea a modo de retroalimentación o la función que los usuarios decidan

Carga de imágenes, multimedia, archivos pdf, word, etc: Permite al usuario cargar imágenes, videos y archivos en formato, pdf, word, excel, power point, etc. Con el fin de compartir el contenido a visualizar por los demás usuarios.

Descargar contenidos: Permite al usuario descargar a su ordenador cualquier tipo de contenido previamente cargado y almacenado en la plataforma.

- **Usuarios del Sistema.**

El componente de contenido está diseñado para ser utilizado por los siguientes usuarios:

Estudiantes: Los estudiantes utilizarán el componente de contenido para acceder al material educativo.

Profesores: Los profesores utilizarán el componente de contenido para crear y administrar el material educativo.

Administradores: Los administradores utilizarán el componente de contenido para gestionar el sistema educativo.

Invitado: Utilizarán el componente de contenido para acceder al material educativo.

Funcionalidad	Administradores	Docente	Invitado	Alumno
Crear Contenidos		✓		
Editar contenidos	✓	✓		
Eliminar contenidos	✓	✓		
Compartir comentario	✓	✓		✓
Visualizar contenidos	✓	✓	✓	✓
Comentar contenidos		✓		✓
Exportar contenidos	✓	✓		✓

- **Restricciones.**

Solo pueden ingresar usuarios que sean estudiantes, invitados, docentes o administradores.

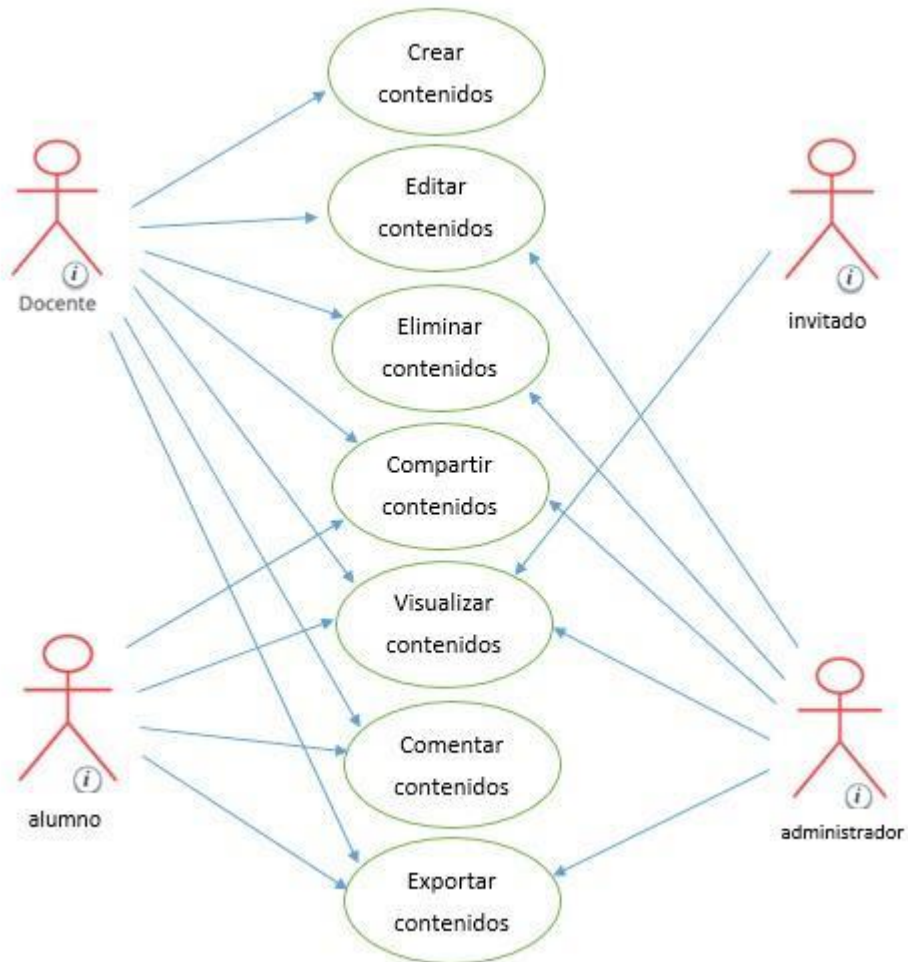
Solo los administradores pueden modificar, actualizar, diseñar o renovar el software.

Los estudiantes solo pueden acceder al contenido, no pueden modificarlo.

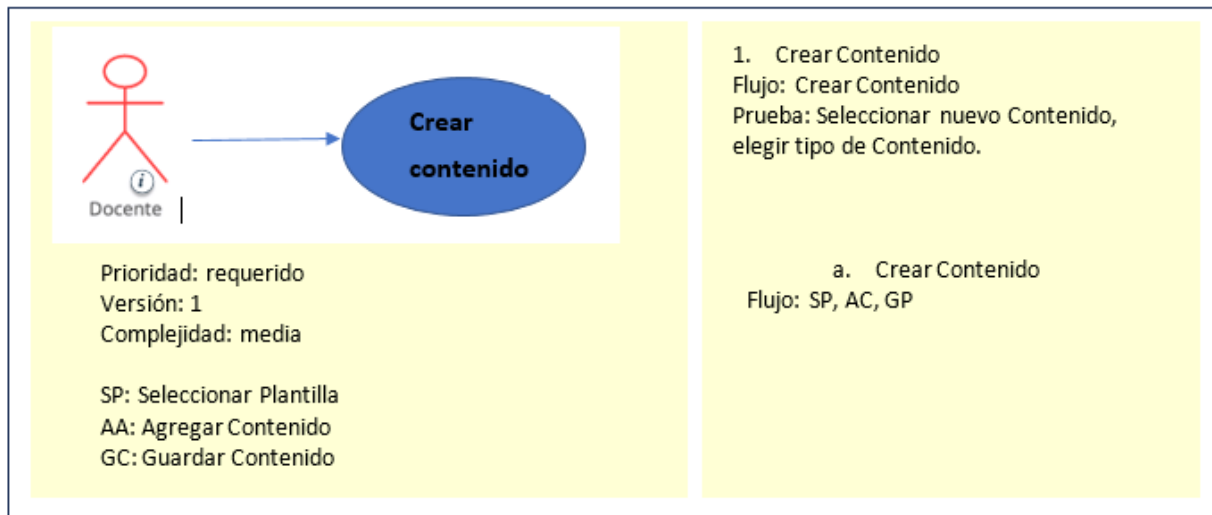
Los profesores solo pueden crear el material educativo y administrarlo, para realizar otra modificación se le debe notificar al administrador.

3. REQUISITOS FUNCIONALES.

- **Casos de Uso.**
 - Descripción detallada de cada caso de uso.



- **Diagramas de Flujo de Casos de Uso**



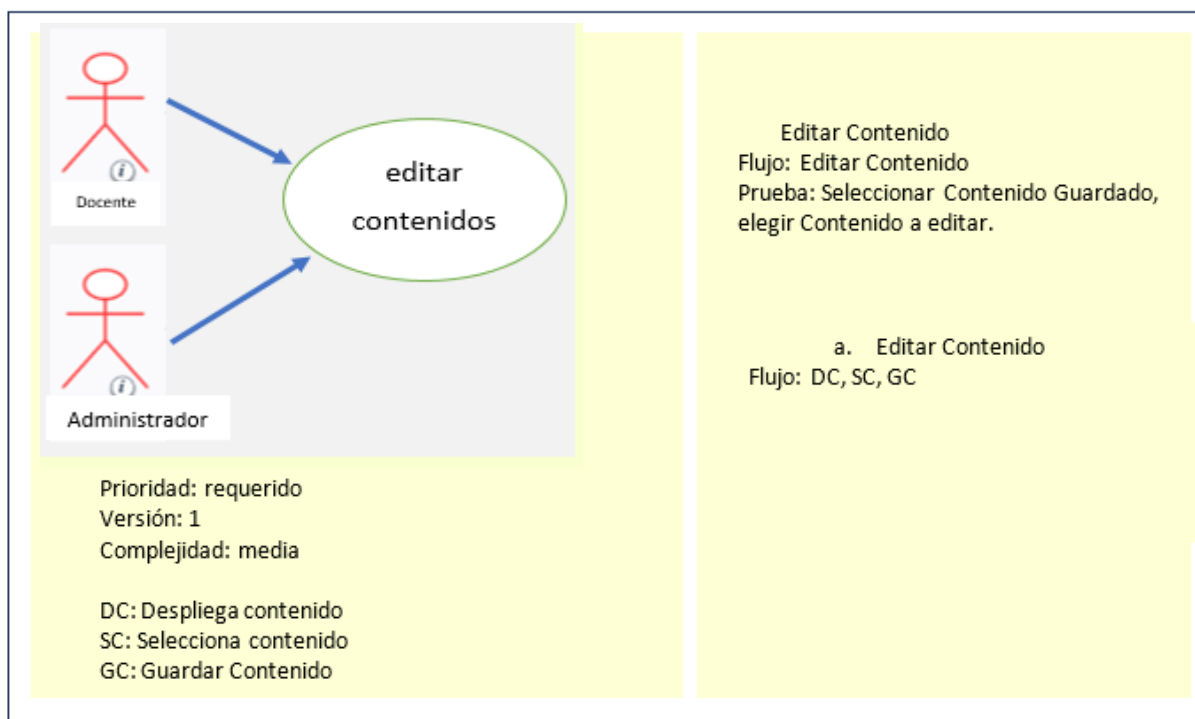
Descripción detallada

CASO No. 1: CREAR CONTENIDO

ID:	CU-1	
Nombre	CREAR CONTENIDO	
Actores	Docente	
Objetivo	Este caso debe permitir crear contenidos	
Urgencia	5	
Esfuerzo	5	
Pre-condiciones	- Debe haberse autenticado de forma correcta en el sistema.	
Flujo Normal	Docente	Sistema
	Selecciona el tipo de contenido a subir	
		Le brinda la opción a elegir de quienes visualizan el contenido
	Selecciona quienes visualizan el contenido	

ID:	CU-1	
Nombre	CREAR CONTENIDO	
Actores	Docente	
Objetivo	Este caso debe permitir crear contenidos	
Urgencia	5	
Esfuerzo	5	
		Le da la opción de elegir el contenido que subirá
	Selecciona el documento a subir	
		Sube el documento
	Guarda el documento	
		Da a visualizar el contenido subido
Flujo alternativo 1	<p>Usuario intenta subir un contenido.</p> <p>El usuario vuelve a intentar con otro tipo de archivo.</p>	<p>El sistema detecta un archivo no compatible.</p> <p>El sistema muestra un mensaje de error y ofrece sugerencias</p>

CASO 2: EDITAR CONTENIDO

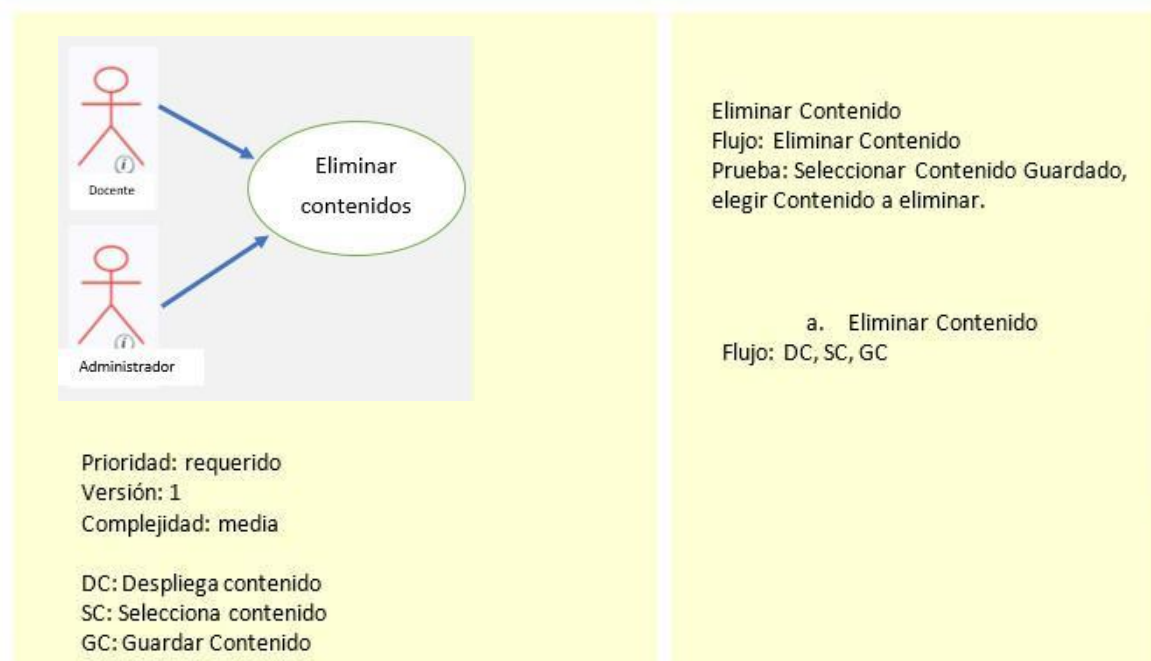


Descripción detallada de cada caso de uso

ID:	CU-2	
Nombre	EDITAR CONTENIDO	
Actores	Docente, Administrador	
Objetivo	Este caso debe permitir editar contenidos	
Urgencia	4	
Esfuerzo	4	
Pre-condiciones	<ul style="list-style-type: none"> - Debe haberse autenticado de forma correcta en el sistema. - Se debe haber creado y guardado cualquier tipo de contenido 	
Flujo Normal	Docente, Administrador	Sistema

	Ingresa al módulo de contenidos	
		Despliega contenido guardado
	Selecciona contenido a visualizar	
		Despliega las opciones disponibles
	Selecciona opción editar contenido	
		Despliega las opciones actualizar contenido
	Registra cambios	
		Guarda contenido
Flujo alternativo 1	<p>Mientras edita, el usuario decide revertir a una versión anterior del contenido.</p> <p>Usuario accede al historial de versiones del contenido.</p> <p>Sistema muestra versiones anteriores.</p> <p>Selecciona la versión específica.</p> <p>Usuario confirma la restauración.</p> <p>Sistema guarda los cambios.</p>	

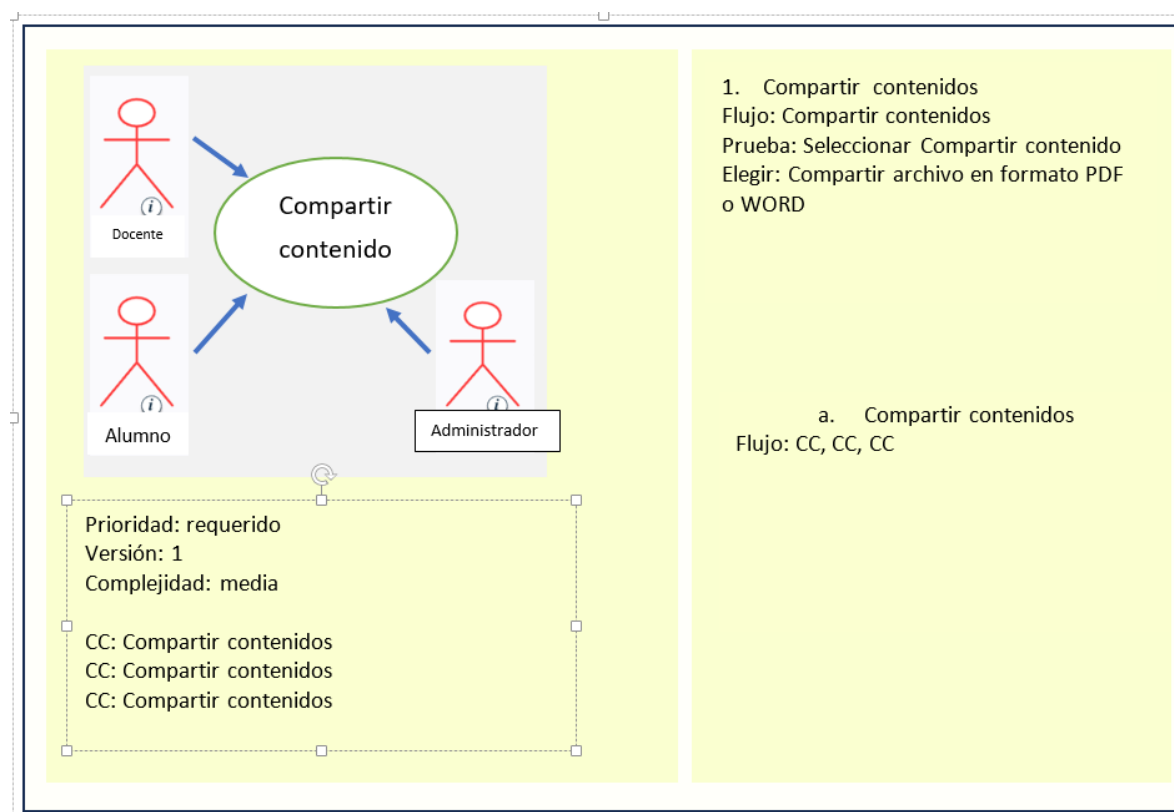
CASO 3: ELIMINAR CONTENIDO



ID:	CU-3	
Nombre	ELIMINAR CONTENIDO	
Actores	Docente, Administrador	
Objetivo	Este caso debe permitir eliminar contenidos	
Urgencia	4	
Esfuerzo	4	
Pre-condiciones	<ul style="list-style-type: none"> - Debe haberse autenticado de forma correcta en el sistema. - Se debe haber creado y guardado cualquier tipo de contenido 	
Flujo Normal	Docente, Administrador	Sistema
	Ingresa al módulo de contenidos	

		Despliega contenido guardado
	Selecciona contenido a visualizar	
		Despliega las opciones disponibles
	Selecciona opción eliminar contenido	
		Despliega la opción confirmar eliminar contenido
	Registra cambios	
		Guarda contenido
Flujo alternativo 1		

CASO No.4 COMPARTIR CONTENIDOS

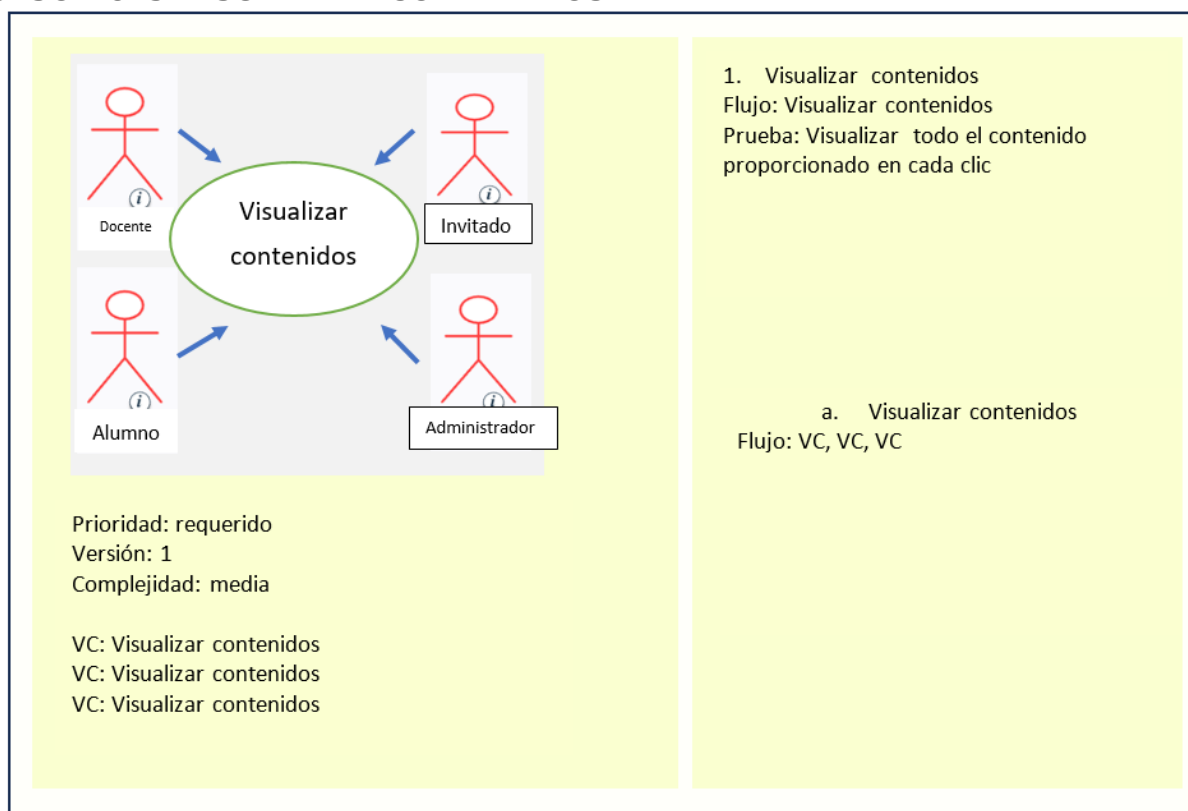


CASO No.4 Compartir contenidos

ID:	CU-4
Nombre	Compartir contenidos
Actores	Docente, Estudiante y administrador
Objetivo	Este caso debe permitir compartir contenidos
Urgencia	5
Esfuerzo	4

Pre-condiciones	- Debe haberse autenticado de forma correcta en el sistema.	
Flujo Normal	Docente, alumno y administrador	Sistema
	Se selecciona el contenido que se quiere compartir.	
		Despliega las opciones disponibles para compartir el contenido, ya sea correo, WhatsApp, etc.
	Selecciona donde se compartirá el contenido	
		Envía el contenido a donde se ha compartido
		Da la notificación que el contenido fue compartido con éxito.

CASO No. 5 VISUALIZAR CONTENIDOS



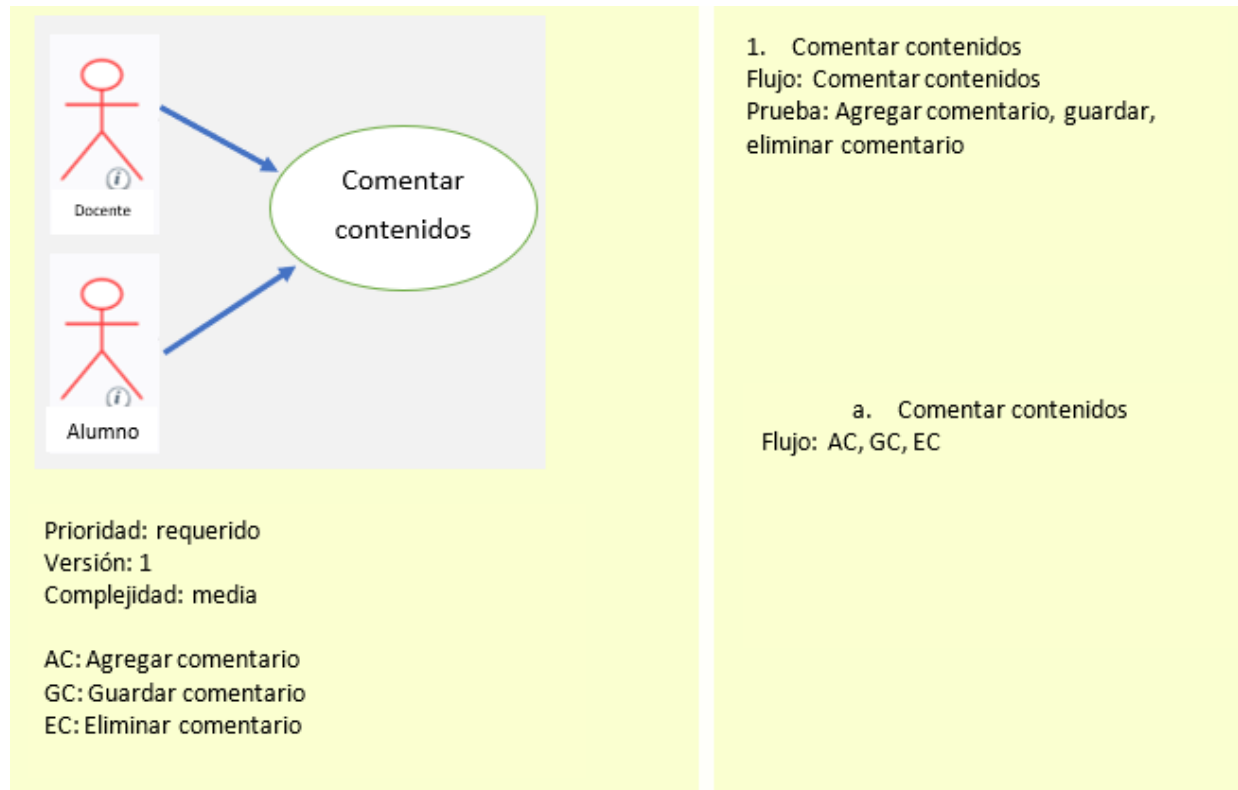
Descripción detallada de cada caso de uso

CASO No. 5 Visualizar contenidos

ID:	CU-5
Nombre	Visualizar contenidos
Actores	Docente, alumno, invitado, administrador
Objetivo	Este caso permite visualizar todos los contenidos
Urgencia	4
Esfuerzo	3
Pre-condiciones	<ul style="list-style-type: none"> - Debe dar la visualización de forma correcta en el sistema. - Debe ofrecer una visualización agradable para todos

Flujo Normal	Docente, alumno, administrador e invitado	Sistema
	El usuario ingresa a la plataforma	
		Despliega en cada clic nuevas ventanas e Interfaz en la pantalla de visualización
	El usuario podrá escoger el contenido a visualizar	
		Brinda la visualización del contenido
	El usuario visualiza el contenido para su interés.	
Flujo alternativo 1	El usuario selecciona la opción modo nocturno.	
		El sistema ajusta a una baja luminosidad los colores del entorno.
Flujo alternativo 2	El usuario deselectiona la opción modo nocturno.	
		El sistema ajusta la luminosidad al estado normal.

Caso 6: COMENTAR CONTENIDO



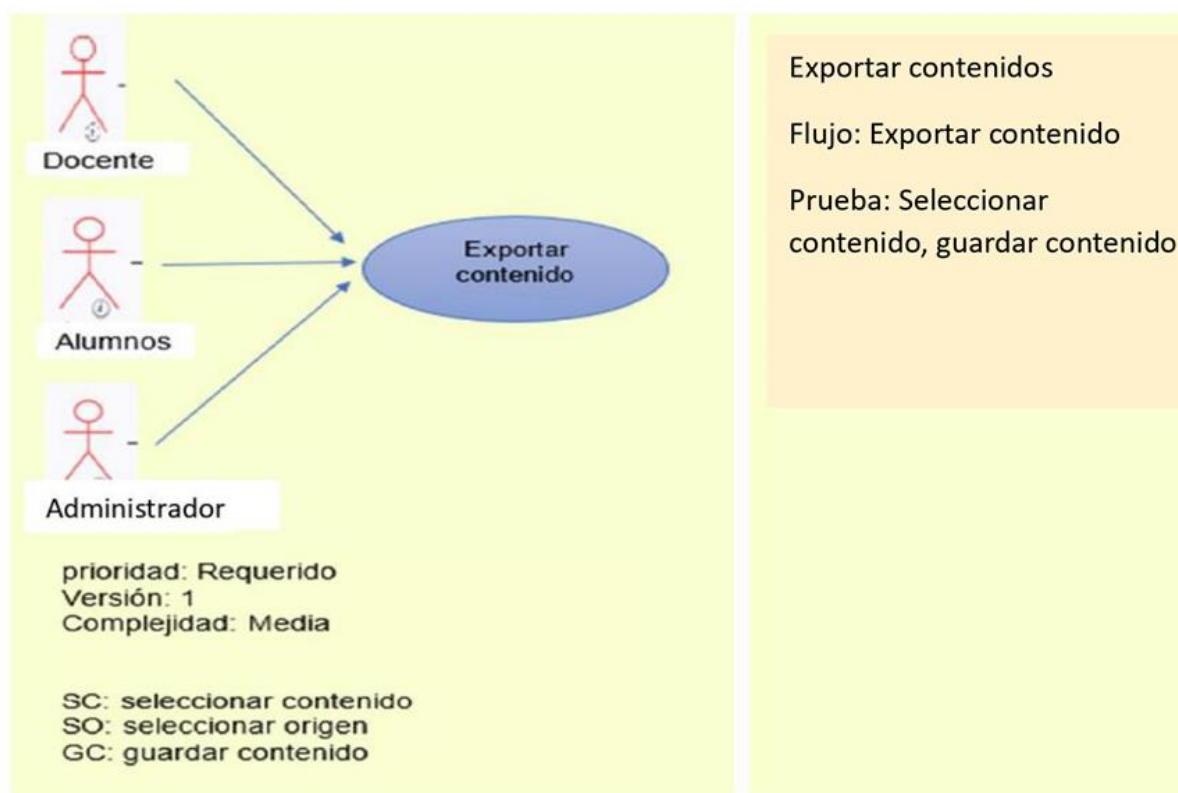
Descripción detallada de cada caso de uso

CASO No. 6 Comentar contenidos

ID:	CU-6
Nombre	Comentar contenidos
Actores	Docente, alumno
Objetivo	Este caso permitirá agregar un comentario a un contenido
Urgencia	5
Esfuerzo	3
Pre-condiciones	- Debe haberse autenticado de forma correcta en el sistema.

	- Debe haber contenidos para comentar	
Flujo Normal	Docente, alumno	Sistema
	Ingresa a contenidos	
		Le brinda la visualización de todos los contenidos
	Selecciona el contenido	
		Abre el contenido seleccionado
	Puede agregar un comentario	
	Puede eliminar su comentario	
	Selecciona comentario guardar	
		Guarda el comentario
Flujo alternativo 1	<p>- El usuario quiere comentar un contenido y dar su opinión.</p> <p>-El sistema le da un aviso que puede usar caracteres limitados.</p> <p>-El usuario se excede de caracteres.</p> <p>-El sistema no permite que haga su comentario satisfactorio</p> <p>-El usuario vuelve a intentar con otro comentario controlando sus caracteres</p>	

CASO No. 7: EXPORTAR CONTENIDOS



Caso No. 7 Exportar contenidos

ID	CU-7
Nombre	Exportar contenidos
Actores	Docente, alumnos, administrador
Objetivo	Este caso debe permitir exportar contenidos de la página en un formato específico
Urgencia	3
Esfuerzo	4
Pre-condiciones	<ul style="list-style-type: none"> - Debe ser usuario con el rol de estudiante, docente o administrador - Deben haber contenidos para exportar

Flujo Normal	Usuario (Docente, estudiante, administrador)	Sistema
	Ingresa a contenidos	
		Le brinda los contenidos disponibles
	Selecciona el contenido	
		Muestra opción de descargar
	Descarga contenido	
		Guarda contenido en la carpeta de descargas

- **Prioridad de Requisitos**

Prioridad de Requerimientos A partir del análisis de requerimientos, funcionalidades y el proceso de design thinking, se concreta la siguiente matriz de prioridad de requerimientos. Para la interpretación se tiene en cuenta la siguiente escala con sus valores.

Eje de Urgencia:

- | | |
|-------------------|-------------|
| - Obligatoria (5) | - Menor (2) |
| - Alta (4) | - Baja (1) |
| - Moderada (3) | |

Eje de Esfuerzo:

- | | |
|----------------|----------------|
| - Muy alto (5) | - Bajo (2) |
| - Alto (4) | - Muy bajo (1) |
| - Medio (3) | |

Urgencia						
Esfuerzo		Baja	Menor	Moderada	Alta	Obligatoria
	Muy alto	5	10	15	20	25
						CU-1 CU-1
	Alto	4	8	12	16	20
				CU-7	CU-2 CU-3	CU-4
	Medio	3	6	9	12	15
					1 CU-5	CU-6
	Bajo	2	4	6	8	10
	Muy bajo	1	2	3	4	5

4. REQUISITOS NO FUNCIONALES.

- **Requisitos de Desempeño.**

Actualización en tiempo real: Las actualizaciones y modificaciones que hagan los administradores, se verán reflejados de manera instantánea en la página.

Navegación rápida: La página debe proporcionar una navegación rápida, mostrando los contenidos al instante, permitiendo al usuario navegar de manera óptima y al momento de que este seleccione un apartado, este se refleja de manera instantánea.

Estructura jerarquizada y entendible: La página proporciona una estructura de contenido jerarquizada a los usuarios (estudiantes) permitiendo que estos puedan entenderla de manera más fácil y rápida, con el fin de que se les facilite utilizarla de manera eficiente sin presentar ninguna dificultad en el proceso.

- **Requisitos de Seguridad.**

1. **Autenticación segura:**

Los usuarios deben autenticarse de manera segura antes de acceder al módulo de contenido.

Debe implementarse un sistema de gestión de contraseñas robusto, incluyendo la necesidad de contraseñas seguras y medidas de protección contra fuerza bruta.

2. **Control de acceso:**

Se deben definir roles y permisos de acceso para garantizar que solo los usuarios autorizados puedan acceder a determinados contenidos y funciones.

Las áreas sensibles, como la administración del contenido, deben tener un acceso restringido y requerir autenticación adicional para tareas administrativas.

3. **Seguridad de la transmisión de datos:** La comunicación entre el usuario y el servidor debe estar cifrada mediante el protocolo HTTPS para proteger la privacidad de los datos durante la transmisión.

4. **Validación de datos de entrada:** Realizar una verificación estricta de los datos de entrada para evitar la introducción de contenido malicioso o inseguro.
5. **Seguridad de archivos y recursos multimedia:** Controlar y restringir el acceso a archivos multimedia y recursos, asegurándonos de que los usuarios no puedan cargar o acceder a contenido malicioso.
6. **Respuesta a incidentes de seguridad:** Establecer un plan de respuesta a incidentes de seguridad para abordar posibles brechas de seguridad y restaurar la integridad del sistema.
7. **Respaldo y recuperación de datos:** Implementar un sistema de respaldo regular y un plan de recuperación de datos para garantizar que los datos críticos puedan restaurarse en caso de pérdida o daño.

- Requisitos de Usabilidad.

- 1. **Interfaz Intuitiva:** La interfaz de usuario del contenido debe ser intuitiva y fácil de usar, permitiendo a los usuarios realizar acciones como dibujar, agregar contenido y colaborar sin dificultad.
- 2. **Compatibilidad con Dispositivos:** El contenido debe ser compatible, se debe adaptar a diferentes tamaños de pantalla. Dispositivos, incluyendo computadoras de escritorio, tabletas y dispositivos móviles
- 3. **Documentación y Ayuda en Línea:** Se debe proporcionar documentación clara y ayuda en línea para los usuarios, incluyendo tutoriales y recursos de soporte.

- Requisitos de Escalabilidad.

Capacidad de manejo de carga: Debe ser capaz de manejar un gran número de usuarios simultáneos y una gran cantidad de solicitudes, lo que significa un diseño eficiente del sistema.

Rendimiento bajo carga: A medida que el tráfico aumenta, el rendimiento de la plataforma no debe degradarse significativamente. Esto puede requerir técnicas como la distribución de la carga, el almacenamiento en caché y la optimización de la base de datos.

Disponibilidad: la plataforma debe estar disponible en todo momento, independientemente del volumen de tráfico. Esto puede implicar tener redundancia en todos los niveles del sistema (como servidores, bases de datos y conexiones a Internet) para garantizar que el sitio web siga funcionando incluso si una parte del sistema falla.

Funcionalidades y datos

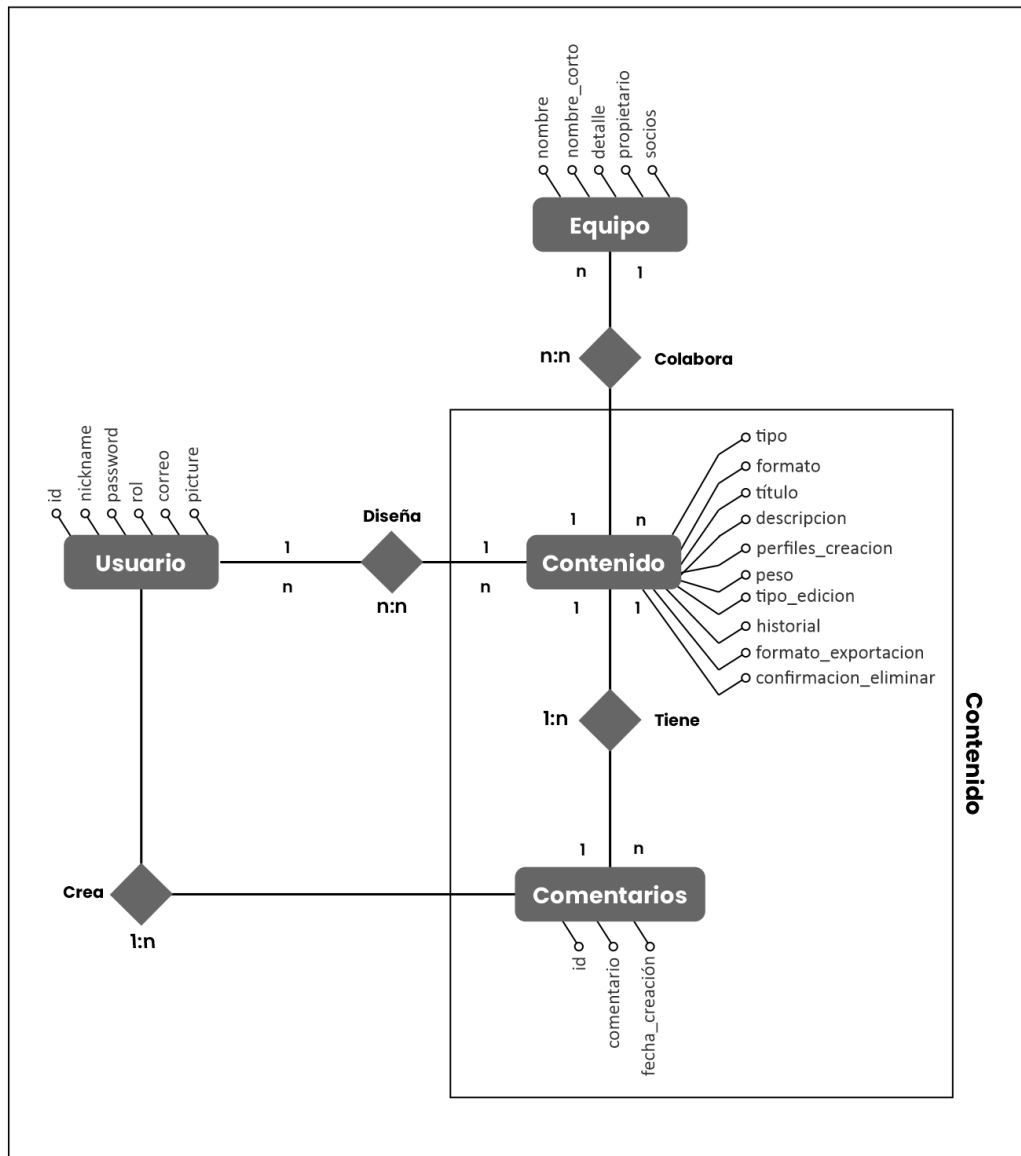
Funcionalidades	Datos
Crear contenido	<ul style="list-style-type: none"> · Tipo de contenido: ["video", "documento", "imagen"] · Formatos admitidos: ["mp4", "dng", "png", "jpg"] · Contenido: "image.jpg" · Título: "El mouse" · Descripción: "Funciones del mouse" · Perfiles de creación: ["admin", "docente", "alumno"] · Peso del documento: "12 mb"
Editar contenido	<ul style="list-style-type: none"> · Tipo de edición: "video" · Historial de cambios: ["1", "2", "3"]
Eliminar contenido	<ul style="list-style-type: none"> · Confirmación de eliminación: ["sí", "no"]
Compartir contenido	<ul style="list-style-type: none"> · Opciones de compartir: ["Facebook", "YouTube"] · Privado: ["sí", "no"]

Visualización de contenido	<ul style="list-style-type: none"> · Compatibilidad con dispositivos: ["computador", "celular"]
Comentarios	<ul style="list-style-type: none"> · Comentarios: ["sí", "no"] · Moderación de comentarios: ["sí", "no"]
Exportar contenido	<ul style="list-style-type: none"> · Formatos de exportación: "mp4"

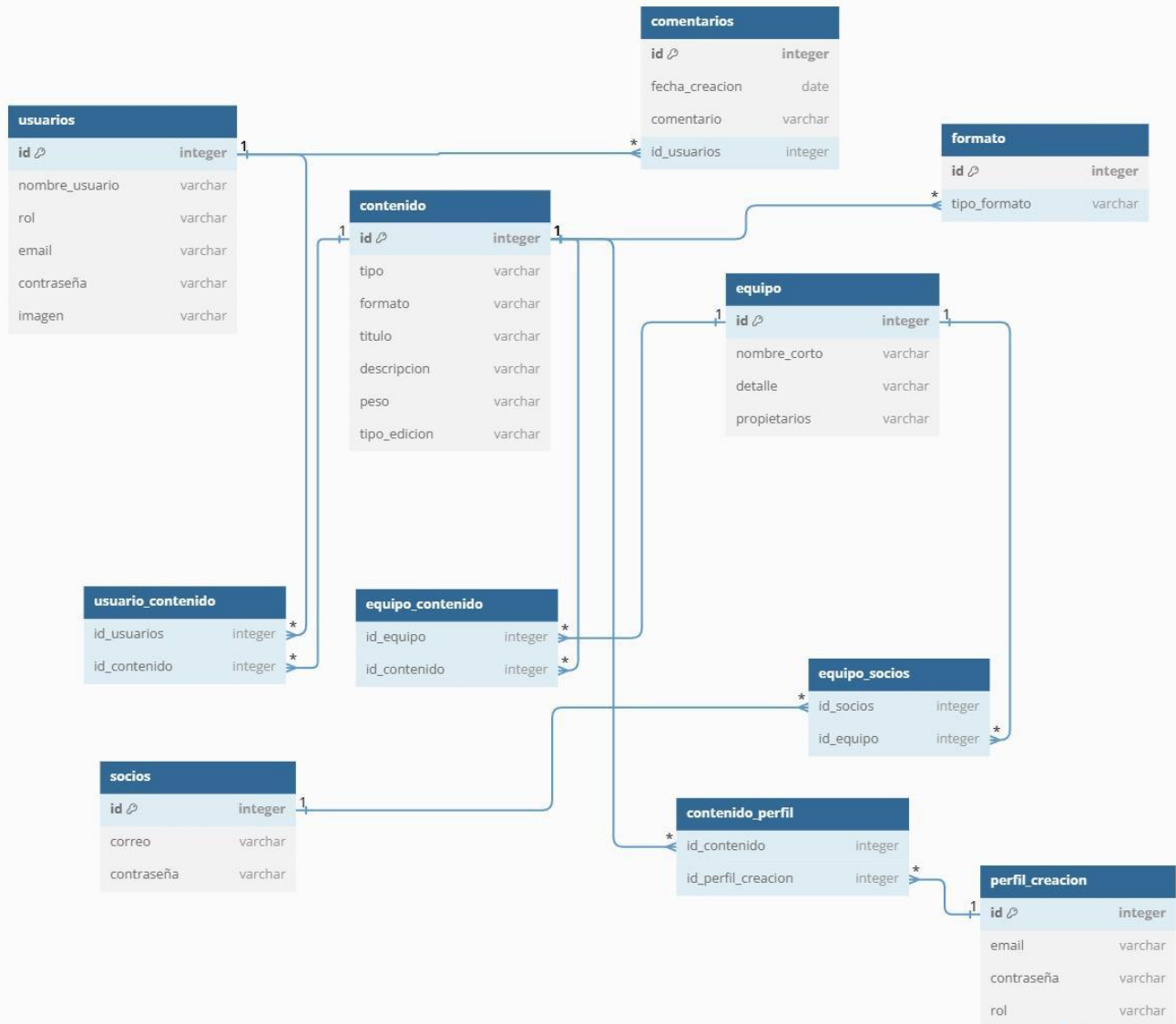
5. MODELADO E/R.

Modelo de Entidad - Relación

Plataforma



Modelo Relacional



Script de modelo relacional.

Table usuarios {

id integer [primary key]
nombre_usuario varchar
rol varchar
email varchar
contraseña varchar
imagen varchar

}

Table contenido {

id integer [primary key]
tipo varchar
formato varchar
titulo varchar
descripcion varchar
peso varchar
tipo_edicion varchar

}

Table comentarios {

id integer [primary key]
fecha_creacion date
comentario varchar
id_usuarios integer

}

Table equipo {

id integer [primary key]
nombre_corto varchar
detalle varchar
propietarios varchar

}

Table usuario_contenido {

id_usuarios integer
id_contenido integer

}

Table equipo_contenido {

id_equipo integer
id_contenido integer

}

Table perfil_creacion {

id integer [primary key]
email varchar
contraseña varchar
rol varchar

}

Table formato {

id integer [primary key]
tipo_formato varchar

}

Table socios {

id integer [primary key]

```

    correo varchar
    contraseña varchar
}
Table equipo_socios {
    id_socios integer
    id_equipo integer
}
Table contenido_perfil {
    id_contenido integer
    id_perfil_creacion integer
}

```

Ref: usuarios.id <
comentarios.id_usuarios

Ref: usuarios.id <
usuario_contenido.id_usuarios

Ref: contenido.id <
usuario_contenido.id_contenido

Ref: equipo.id <
equipo_contenido.id_equipo

Ref: contenido.id <
equipo_contenido.id_contenido

Ref: equipo.id <
equipo_socios.id_equipo

Ref: socios.id <
equipo_socios.id_socios

Ref: contenido.id <
contenido_perfil.id_contenido

Ref: perfil_creacion.id <
contenido_perfil.id_perfil_creacion

Ref: contenido.id <
formato.tipo_formato

Descripción de entidades y relaciones.

ENTIDADES

1. User

Descripción: Un usuario es cualquier persona que interactúa con el sistema. Puede ser un estudiante, profesor o administrador.

Atributos: ID, Nickname, Password, Role, Email, Picture.

2. Content

Descripción: Representa el material educativo creado y compartido en la plataforma.

Atributos: ID, Type, Format, Title, Description, Size, Edition_Type.

3. Comment

Descripción: Un comentario es una anotación que un usuario deja en relación a un contenido. Los comentarios son una herramienta clave para la interacción y retroalimentación en la plataforma.

Atributos: ID, Date_Creation, Comment, ID_User.

4. Team

Descripción: Un equipo es un grupo de usuarios que colaboran juntos dentro de la plataforma.

Atributos: ID, Name, Short_Name, Detail, Owner.

5. Format

Descripción: Representa los diferentes formatos en los que se puede presentar el contenido educativo.

Atributos: ID, Type_Format.

6. Profile_Creation

Descripción: Representa la información necesaria para la creación de un perfil de usuario en la plataforma.

Atributos: ID, Email, Password, Role.

7. Partners

Descripción: Representa las entidades colaboradoras que pueden estar asociadas a los equipos dentro de la plataforma.

Atributos: ID, Email, Password.

RELACIONES

usuarios.id <-> comentarios.id_usuarios

Un usuario puede tener múltiples comentarios.

usuarios.id <-> usuario_contenido.id_usuarios

Un usuario puede tener múltiples contenidos asociados.

contenido.id <-> usuario_contenido.id_contenido

Un contenido puede estar asociado a múltiples usuarios.

equipo.id <-> equipo_contenido.id_equipo

Un equipo puede tener múltiples contenidos asociados.

contenido.id <-> equipo_contenido.id_contenido

Un contenido puede estar asociado a múltiples equipos.

equipo.id <-> equipo_socios.id_equipo

Un equipo puede tener múltiples socios asociados.

socios.id <-> equipo_socios.id_socios

Un socio puede estar asociado a múltiples equipos.

contenido.id <-> contenido_perfil.id_contenido

Un contenido puede estar asociado a múltiples perfiles de creación.

perfil_creacion.id <-> contenido_perfil.id_perfil_creacion

Un perfil de creación puede tener múltiples contenidos asociados.

contenido.id <-> formato.tipo_formato

Un contenido puede estar asociado a un tipo de formato.

Reglas de Integridad Referencial**Entidad: Contenido**

Clave Primaria: ID (IDContenido) - Un campo único para cada contenido.

Clave Externa: Clave externa relacionada con la entidad "Formato" para asociar formatos específicos con contenidos.

Entidad: User

Clave Primaria: ID (IDUsuario) - Un campo único para cada usuario.

Clave Externa: N/A

Entidad: Comentario

Clave Primaria: ID (IDComentario) - Un campo único para cada comentario.

Clave Externa: Clave externa relacionada con la entidad "User" (IDUsuario) para asociar comentarios específicos con usuarios.

Entidad: Team

Clave Primaria: ID (IDEquipo) - Un campo único para cada equipo.

Clave Externa: N/A

Entidad: Formato

Clave Primaria: ID (IDFormato) - Un campo único para cada formato.

Clave Externa: N/A

Entidad: Perfil_Creacion

Clave Primaria: ID (IDPerfilCreacion) - Un campo único para cada perfil de creación.

Clave Externa: N/A

Entidad: Socios

Clave Primaria: ID (IDSocio) - Un campo único para cada socio.

Clave Externa: N/A

Colecciones (NoSLQ)

Users{

"_id": ObjectId(),
"nickname": string,
"password": string,
"role": string,
"email": string,
"picture": image

}

Content{

"_id": ObjectId(),
"type": string,
"format": string,
"title": string,
"description": string,
"size": string,
"edition_type": string

}

Comments{

"_id": ObjectId(),
"date_creation": Date,
"comment": string,
"id_user": ObjectId()

}

Teams{

"_id": ObjectId(),
"name": string,
"short_name": string,
"detail": string,
"owner": string

}

Formats{

"_id": ObjectId(),
"type_format": string

}

Profile_Creation{

"_id": ObjectId(),
"email": string,
"password": string,
"role": string

}

Partners {

"_id": ObjectId(),
"email": string,
"password": string

}

6. ANEXOS.

- Diagramas adicionales
- Referencias

Etapas 2: Persistencia de Datos con Backend

7. Introducción

La segunda etapa de persistencia de datos con backend desempeña un papel fundamental en el desarrollo de nuestro software educativo orientado a la creación de presentaciones. En esta fase, nos adentramos en el corazón del sistema, donde se conciben y gestionan los contenidos que serán la columna vertebral de la experiencia de aprendizaje. En esta sección introductoria, exploraremos el propósito y el alcance de esta etapa crítica, delineando claramente los objetivos que buscamos alcanzar y los límites que definirán nuestra labor.

Propósito de la Etapa

El propósito fundamental de esta etapa de persistencia de datos es la creación, gestión y almacenamiento eficiente de los contenidos educativos que enriquecerán la experiencia de nuestros usuarios. Aquí se forjarán las bases para brindar un entorno de aprendizaje interactivo y personalizado. La etapa de persistencia de datos es el eslabón clave que garantiza la disponibilidad, accesibilidad y actualización de los recursos pedagógicos, lo que contribuirá significativamente al éxito de nuestro software educativo.

Alcance de la Etapa

En cuanto al alcance, esta etapa abarcará la definición de la estructura de datos, la implementación de un sistema de gestión de contenidos (CMS) adaptado a las necesidades educativas, la interacción con los usuarios para la creación y modificación de contenidos, y la integración con otros componentes del software, como la interfaz de usuario y las funcionalidades de seguimiento del progreso del estudiante. Asimismo, abordaremos la seguridad, la escalabilidad y la eficiencia en el almacenamiento de datos para garantizar un funcionamiento óptimo.

Con esta visión general, estamos listos para adentrarnos en los detalles de la segunda etapa de persistencia de datos con backend y trabajar en conjunto para construir una plataforma educativa de calidad que cumpla con nuestras metas y expectativas.

8. Diseño de la Arquitectura de Backend

Descripción de la Arquitectura Propuesta

La arquitectura del backend se basa en el marco de trabajo Nest.js, que sigue el patrón Modelo-Vista-Controlador (MVC) y proporciona una estructura organizada y escalable para el desarrollo.

Capa de Presentación: En esta capa, Nest.js maneja las solicitudes HTTP y las rutas a los controladores correspondientes, permitiendo la interacción con el cliente.

Capa de Lógica de Negocio: Los servicios de Nest.js contienen la lógica de la aplicación, como la gestión de usuarios, contenidos y categorías.

Capa de Datos: Aquí se gestionan las operaciones de lectura y escritura de datos en la base de datos, con modelos y esquemas definidos.

Tecnologías Específicas de Nest.js:

- **Decoradores:** Utilizados para definir rutas y metadatos en controladores y servicios.
- **Módulos:** Organizan y encapsulan componentes relacionados para modularidad y reutilización del código.
- **Inyección de Dependencias:** Proporciona servicios y componentes a las clases que los necesitan.
- **Middleware:** Agrega funcionalidades adicionales, como autenticación y autorización, a las rutas.

Escalabilidad: Nest.js permite manejar un mayor número de solicitudes con balanceadores de carga y escalabilidad en múltiples instancias.

Caché: Implementación de una capa de caché para mejorar el rendimiento y reducir la carga en la base de datos.

Documentación de la API: Generación automática de documentación de API basada en controladores y rutas.

Monitorización y Registro: Uso de herramientas para supervisar el rendimiento y detectar problemas en tiempo real.

Seguridad: Aplicación de prácticas sólidas de seguridad, incluyendo protección contra ataques, autenticación segura y autorización adecuada.

- Componentes del backend.
- Diagramas de arquitectura.

9. ELECCIÓN DE LA BASE DE DATOS.

10. IMPLEMENTACIÓN DEL BACKEND.

/11. CONEXIÓN A LA BASE DE DATOS.

12. PRUEBAS DEL BACKEND

Etapas 3: Consumo de datos y desarrollo frontend.

13. INTRODUCCIÓN

14. CREACIÓN DE LA INTERFAZ DE USUARIO (UI)

15. PROGRAMACIÓN FRONTEND CON JAVASCRIPT (JS)

16. CONSUMO DE DATOS DESDE EL BACKEND.

17. INTERACCIÓN USUARIO-INTERFAZ.

18. PRUEBAS Y DEPURACIÓN DEL FRONTEND.

19. IMPLEMENTACIÓN DE LA LÓGICA DE NEGOCIO EN EL FRONTEND.

20. INTEGRACIÓN CON EL BACKEND.