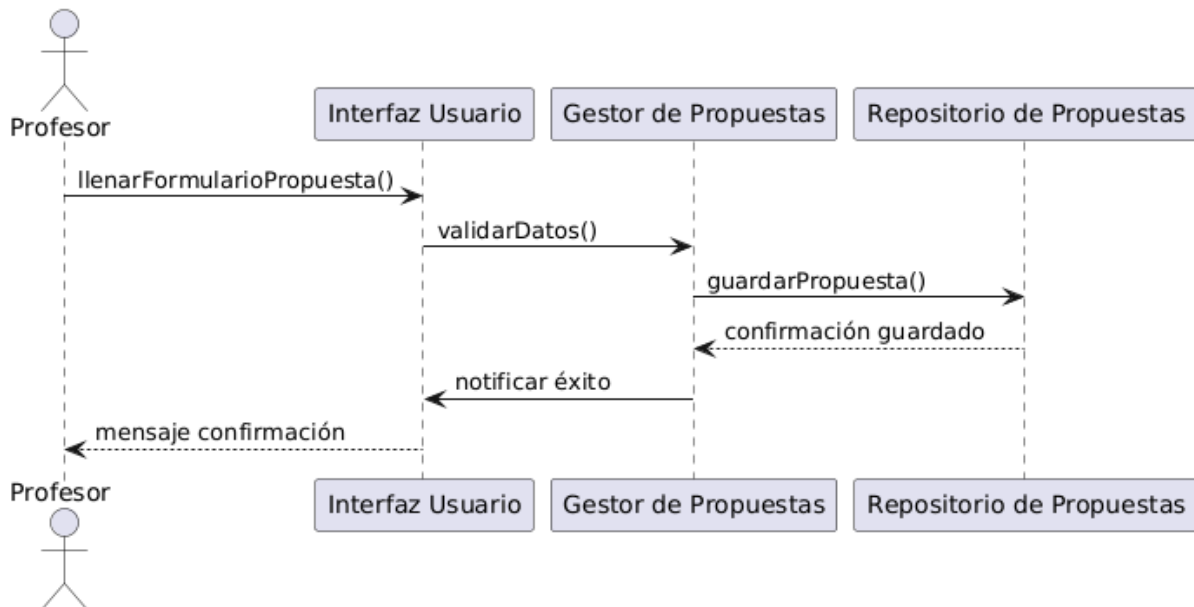


Diagramas de secuencia

1. Enviar propuesta de cambio curricular: Este diagrama representa cómo un **profesor o director de programa** interactúa con el sistema para enviar una propuesta de cambio.

Diagrama de Secuencia - Enviar Propuesta de Cambio Curricular



Elementos del Diagrama

1. Actores:
 - Profesor: Representa al usuario que inicia la acción de enviar una propuesta.
2. Objetos:
 - Interfaz Usuario: La interfaz a través de la cual el profesor interactúa con el sistema.
 - Gestor de Propuestas: Componente que maneja la lógica de negocio para gestionar propuestas curriculares.
 - Repositorio de Propuestas: Base de datos o almacenamiento donde se guardan las propuestas.

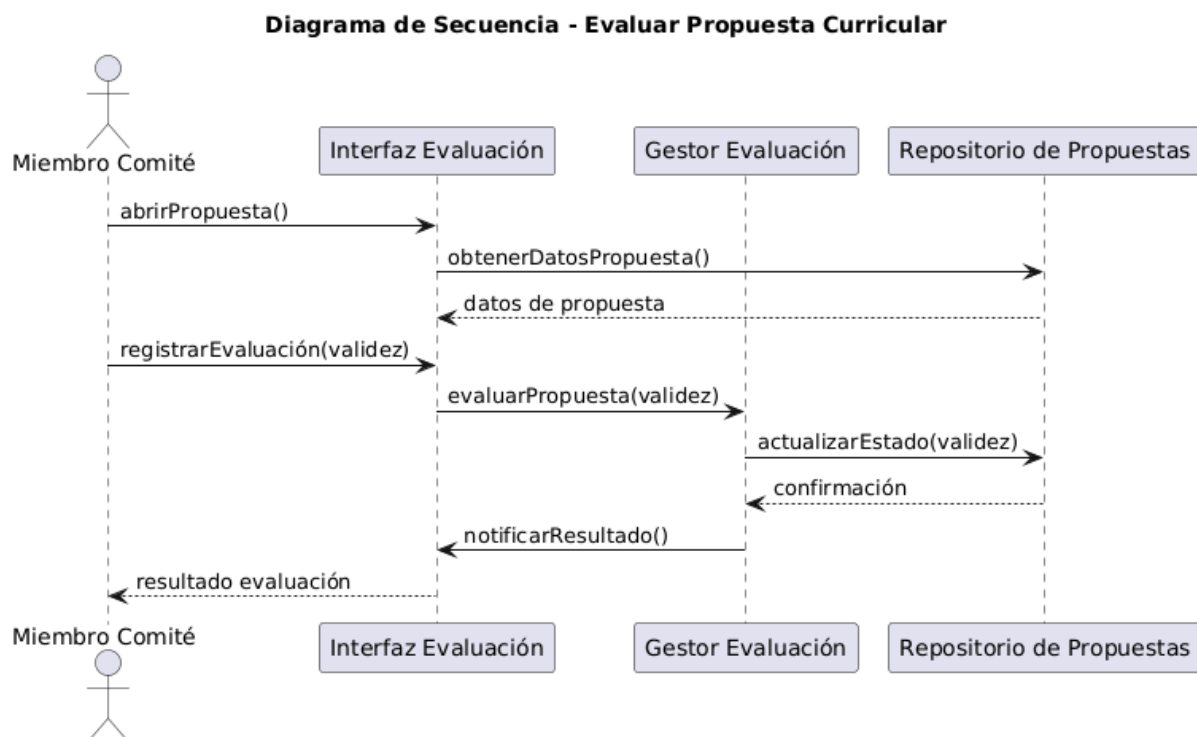
Flujo del Proceso

1. Llenar Formulario de Propuesta:
 - El Profesor inicia el proceso llenando un formulario que forma parte de la Interfaz Usuario. Esta acción se representa con `llenarFormularioPropuesta()`.
2. Validar Datos:
 - Una vez que el profesor ha llenado el formulario, el sistema llama a `validarDatos()` dentro del Gestor de Propuestas. Este método se encarga de verificar que la información proporcionada en el formulario sea correcta y cumpla con los requisitos necesarios.
3. Guardar Propuesta:
 - Si la validación es exitosa (es decir, los datos son correctos), el sistema ejecuta `guardarPropuesta()` para almacenar la propuesta en el Repositorio de Propuestas.
4. Confirmación Guardada:

- Después de que la propuesta ha sido guardada correctamente, el sistema envía una respuesta que indica que la acción fue exitosa. Esta respuesta puede ser representada como un mensaje de confirmación.
5. Notificar Éxito:
- Finalmente, se envía un mensaje de confirmación al Profesor indicando que su propuesta ha sido enviada y está guardada en el sistema.

2. Diagrama de Secuencia – Evaluar propuesta

Interacción del **comité o director de escuela** con el sistema para evaluar una propuesta recibida.



Elementos del Diagrama

1. Miembro Comité: Representa al usuario que va a realizar la evaluación de la propuesta.
2. Interfaz Evaluación: Es el componente a través del cual el miembro del comité interactúa con el sistema.
3. Gestor Evaluación: Se encarga de gestionar el proceso de evaluación.
4. Repositorio de Propuestas: Almacena las propuestas curriculares que se van a evaluar.

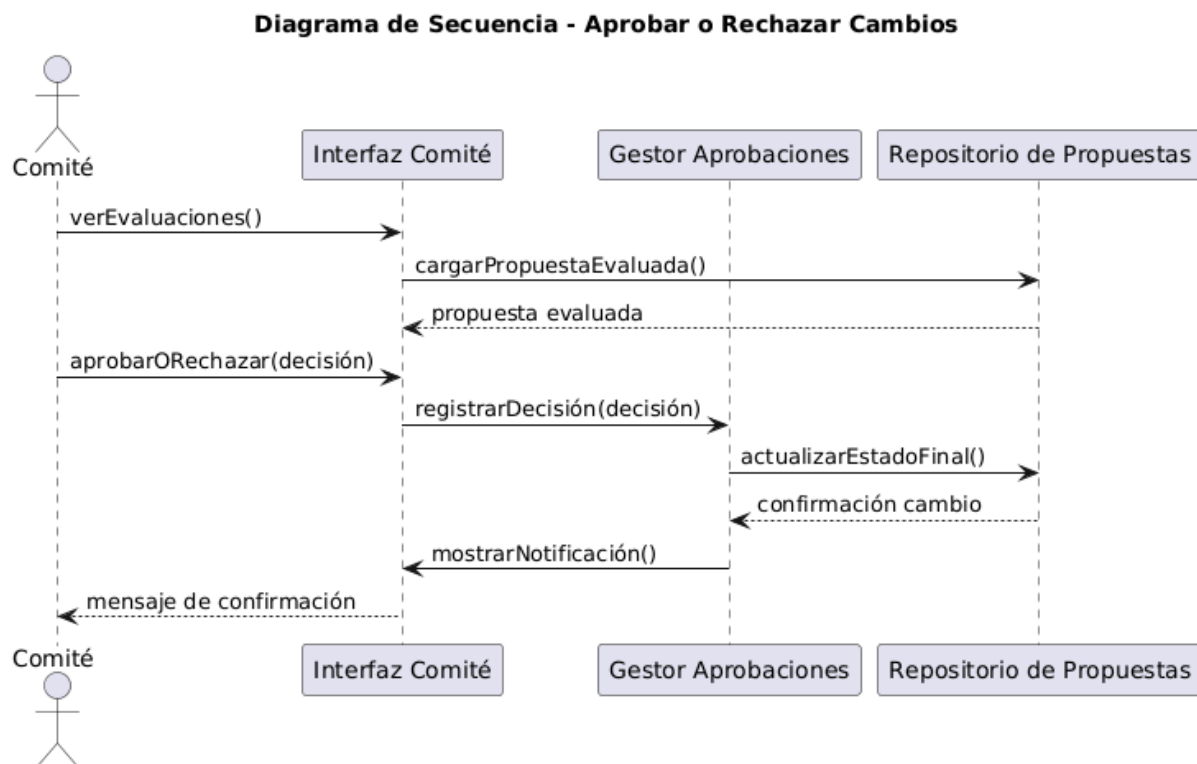
Flujo del Proceso

1. `abrirPropuesta()`:
 - El miembro del comité inicia el proceso abriendo una propuesta desde la interfaz.
2. `obtenerDatosPropuesta()`:
 - El gestor de evaluación solicita los datos relacionados con la propuesta al repositorio.
3. `datos de propuesta`:
 - Se devuelve la información específica de la propuesta solicitada.
4. `registrarEvaluación(validez)`:

- El miembro del comité registra la evaluación realizada, indicando si es válida o no.
- 5. evaluarPropuesta(validéz):
 - Se evalúa la propuesta basada en la validez que ha registrado el miembro del comité.
- 6. actualizarEstado(validéz):
 - El estado de la evaluación se actualiza en el repositorio de propuestas con la validez otorgada.
- 7. confirmación:
 - Se confirma que el estado ha sido actualizado correctamente.
- 8. notificarResultado():
 - Finalmente, se notifica el resultado de la evaluación, informando al miembro del comité de la finalización del proceso.

3. Diagrama de Secuencia – Aprobar / Rechazar Cambios

Esto ocurre luego de la evaluación. El comité decide si aprueba o rechaza la propuesta.



Elementos del Diagrama

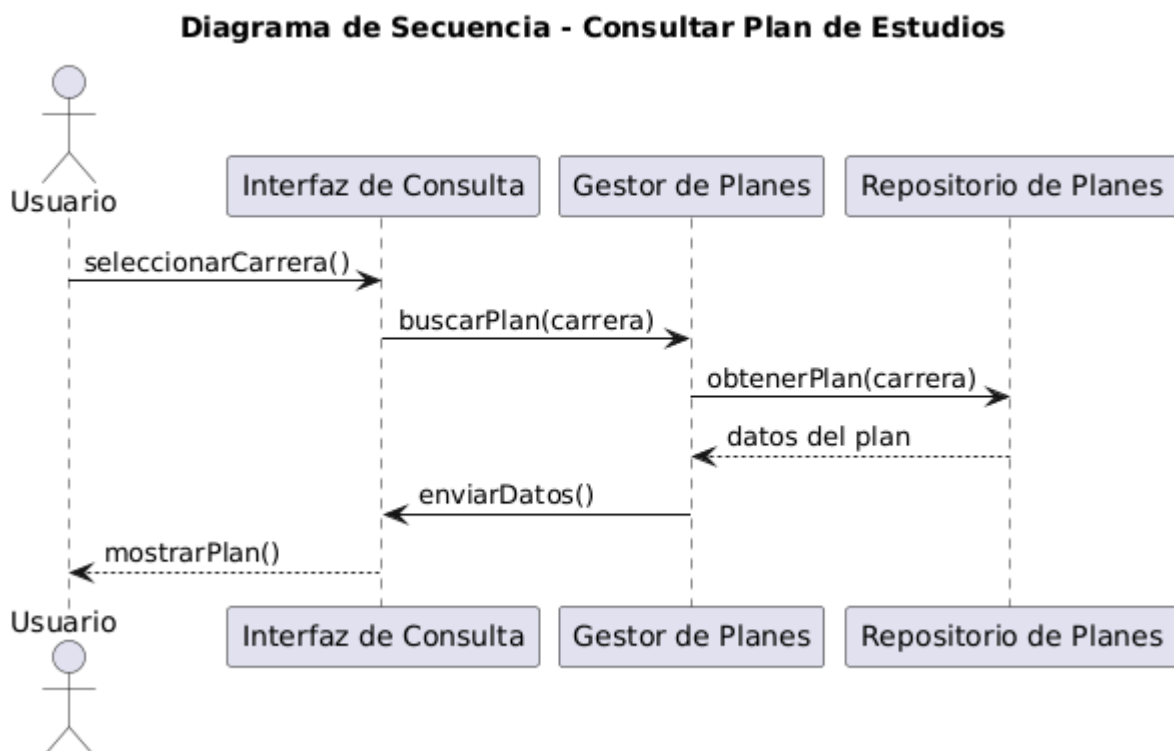
1. Comité: Representa al usuario del comité que gestionará la aprobación o rechazo.
2. Interfaz Comité: Es el componente mediante el cual el comité interactúa con el sistema.
3. Gestor Aprobaciones: Se encarga de gestionar el proceso de revisión de las propuestas.
4. Repositorio de Propuestas: Almacena las propuestas que fueron evaluadas y están listas para ser aprobadas o rechazadas.

Flujo del Proceso

1. verEvaluaciones(): El miembro del comité solicita ver las evaluaciones disponibles a través de la interfaz.
2. cargarPropuestaEvaluada(): El gestor de aprobaciones carga la propuesta que ha sido evaluada y está lista para su decisión.
3. propuesta evaluada:
 - Se muestra la propuesta que ha sido evaluada al comité.
4. aprobarORchazar(decisión): El miembro del comité toma una decisión sobre la propuesta, ya sea aprobarla o rechazarla.
5. registrarDecisión(decisión): Se registra la decisión tomada por el comité en el sistema.
6. actualizarEstadoFinal(): El gestor de aprobaciones actualiza el estado final de la propuesta en el repositorio, reflejando la decisión tomada.
7. confirmación cambio: Se envía una confirmación de que el estado de la propuesta ha sido actualizado correctamente.
8. mostrarNotificación(): Finalmente, se muestra una notificación al miembro del comité informando sobre la confirmación del cambio.

4. Diagrama de Secuencia – Consultar plan de estudios

Tanto profesores como estudiantes consultan el plan actual.



Elementos del Diagrama

1. Usuario: Representa al estudiante o persona que está consultando el plan de estudios.
2. Interfaz de Consulta: Es la pantalla o aplicación a través de la cual el usuario interactúa para obtener información.

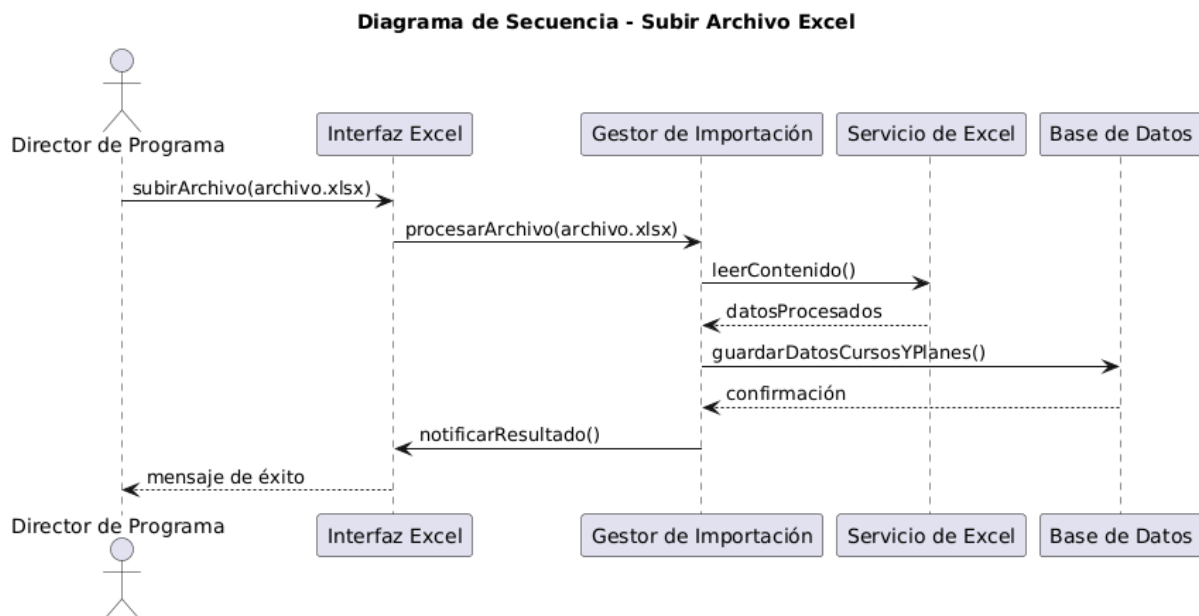
3. Gestor de Planes: Se encarga de gestionar los planes de estudios, actuando como intermediario entre la interfaz y el repositorio.
4. Repositorio de Planes: Almacena la información sobre los planes de estudios disponibles.

Flujo del Proceso

1. seleccionarCarrera():
 - El usuario elige la carrera que desea consultar desde la interfaz de consulta.
2. buscarPlan(carrera):
 - El gestor de planes recibe la carrera seleccionada y busca el plan correspondiente.
3. obtenerPlan(carrera):
 - A través del repositorio de planes, se obtiene la información específica del plan para la carrera seleccionada.
4. datos del plan:
 - Se envían los datos del plan de estudios al gestor de planes.
5. enviarDatos():
 - El gestor de planes envía los datos del plan a la interfaz de consulta.
6. mostrarPlan():
 - Finalmente, la interfaz de consulta presenta el plan de estudios al usuario.

5. Diagrama de Secuencia – Subir archivo Excel

Cuando el director de programa sube un archivo Excel con cursos y planos.



Elementos del Diagrama

1. Director de Programa: Es el usuario que sube el archivo Excel al sistema.
2. Interfaz Excel: Es la aplicación mediante la cual el director interactúa para subir el archivo.

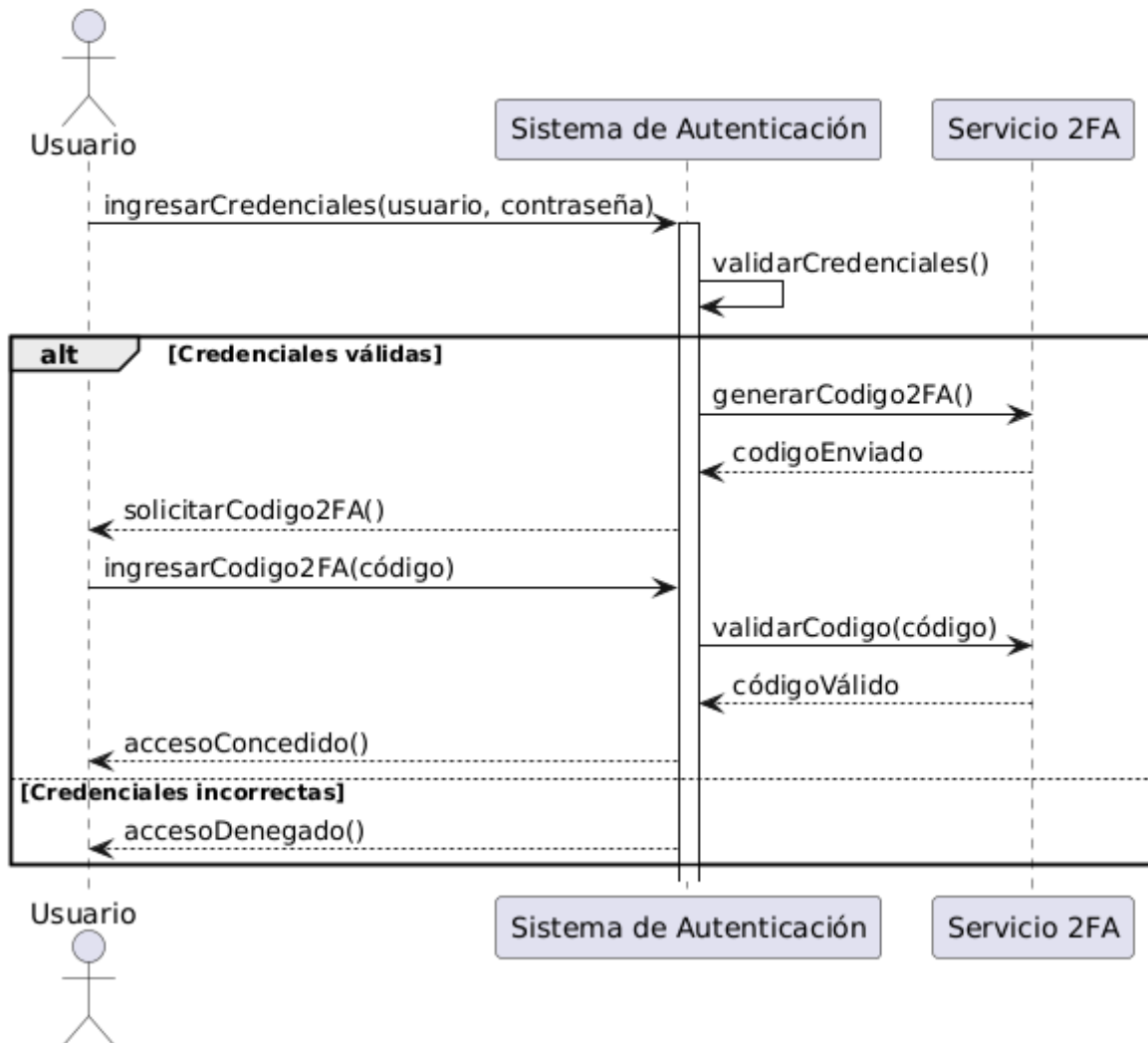
3. Gestor de Importación: Maneja la lógica de importación y procesamiento de archivos.
4. Servicio de Excel: Se encarga de las operaciones relacionadas con la manipulación del contenido del archivo Excel.
5. Base de Datos: Almacena los datos procesados una vez que se han importado correctamente.

Flujo del Proceso

1. subirArchivo(archivo.xlsx):
 - El director de programa inicia el proceso seleccionando el archivo Excel que desea subir desde la interfaz.
2. procesarArchivo(archivo.xlsx):
 - El gestor de importación recibe el archivo y llama a la función para procesarlo.
3. leerContenido():
 - El servicio de Excel lee el contenido del archivo Excel.
4. datosProcesados:
 - Se generan y procesan los datos leídos del archivo.
5. guardarDatosCursosYPlanes():
 - Los datos procesados se envían a la base de datos para ser almacenados.
6. confirmación:
 - La base de datos devuelve una confirmación de que los datos se han guardado correctamente.
7. notificarResultado():
 - El gestor de importación notifica a la interfaz Excel sobre el resultado del proceso.
8. mensaje de éxito:
 - Finalmente, se muestra un mensaje al director de programa en la interfaz, confirmando que el archivo se ha subido y procesado exitosamente.

SPRINT 2

Inicio de sesión con autenticación de dos factores



1. Inicio de sesión con autenticación de dos factores (2FA)

Participantes:

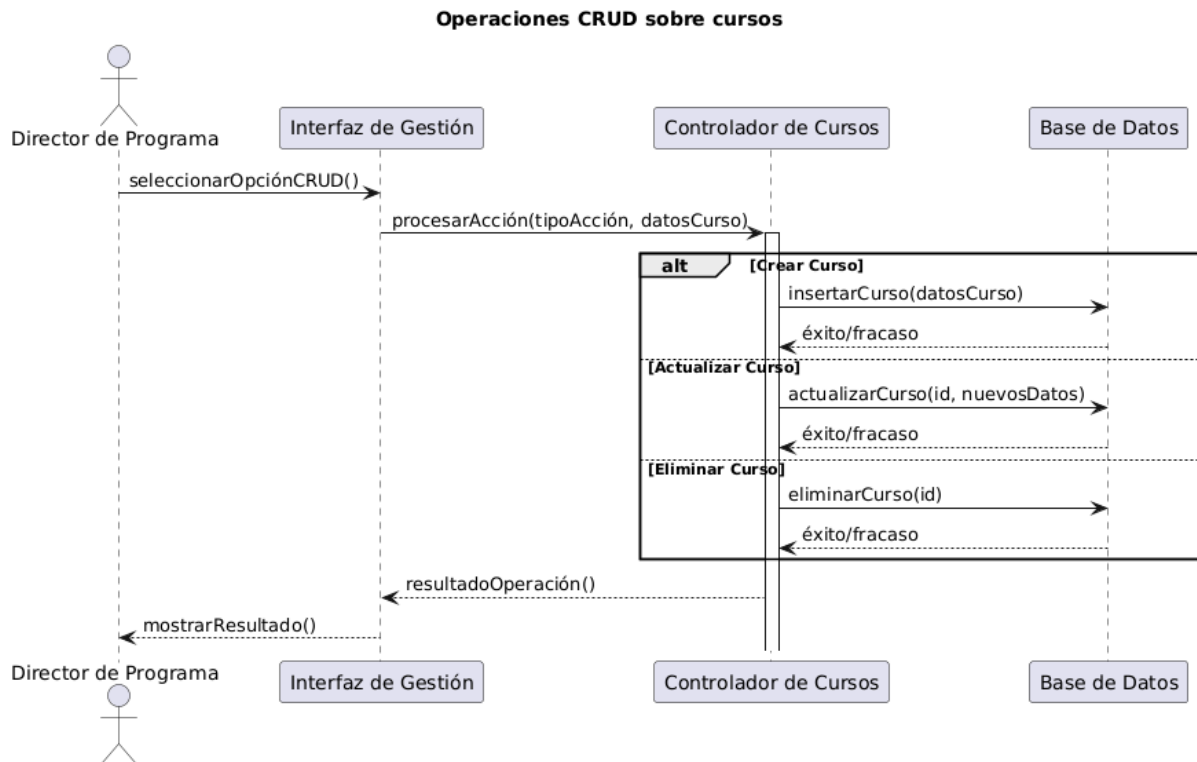
- **Usuario:** Persona que intenta iniciar sesión.
- **Sistema de Autenticación:** Verifica credenciales y coordina la autenticación 2FA.
- **Servicio 2FA:** Encargado de generar y validar códigos temporales.

Descripción:

1. El usuario ingresa sus credenciales al sistema.
2. El sistema valida internamente las credenciales.
3. Si son válidos, se genera un código de autenticación y se envía al usuario mediante el **Servicio 2FA**.
4. El usuario introduce el código.
5. El sistema valida el código con el servicio.

- Si es correcto, el acceso es concedido. Si no, es denegado.

Objetivo: Asegurar que sólo usuarios legítimos accedan al sistema, incluso si sus contraseñas han sido comprometidas.



2. Operaciones CRUD sobre cursos

Participantes:

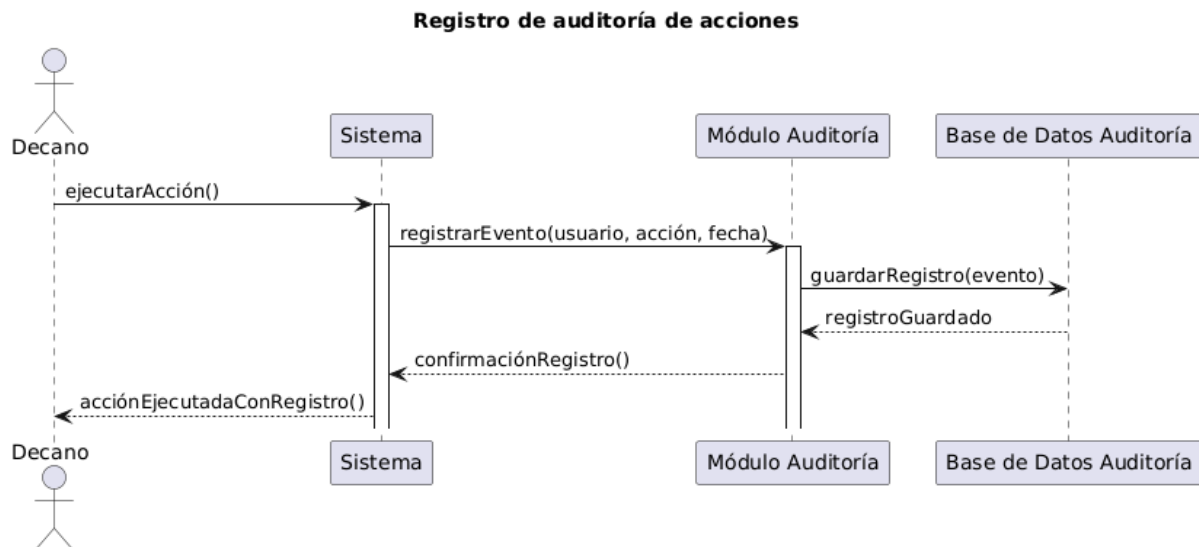
- **Director de Programa:** Usuario con permisos para gestionar cursos.
- **Interfaz de Gestión:** Aplicación web o sistema desde donde se selecciona la acción.
- **Controlador de Cursos:** Componente lógico que coordina las acciones.
- **Base de datos:** Donde se almacenan los cursos.

Descripción:

1. El Director selecciona una operación desde la interfaz.
2. La interfaz envía la solicitud al controlador con los datos.
3. El controlador ejecuta una de las siguientes acciones:
 - **Crear:** Inserta un nuevo curso.
 - **Actualizar:** Modifica un curso existente.
 - **Eliminar:** Borra un curso.

4. La base de datos responde con el resultado.
5. El sistema informa al usuario del resultado.

Objetivo: Administrar la información de los cursos de forma estructurada y segura.



3. Registro de auditoría de acciones

Participantes:

- **Decano:** Actor que realiza acciones dentro del sistema .Actor que realiza acciones dentro del sistema.
- **Sistema:** Captura eventos generados por los usuarios.
- **Módulo Auditoría:** Encargado de almacenar los registros .Encargado de almacenar los registros.
- **Base de Datos de Auditoría:** Almacén especializado en logs o eventos .Almacén especializado en logs o eventos.

Descripción:

1. El decano realiza una acción (como editar un curso).
2. El sistema, además de ejecutar la acción, solicita al módulo de auditoría que registre el evento.
3. El módulo de auditoría guarda los datos (usuario, acción, fecha).
4. Confirma el almacenamiento al sistema. 5
5. El sistema informa al decano que la acción fue ejecutada con registro.

Objetivo: Asegurar trazabilidad, transparencia y cumplimiento normativo