

Trabajo Práctico 2 - Parte 2

Diagramas de Secuencia

Objetivo

A partir de los casos de uso que cada grupo definió en Ingeniería de Software I, y considerando las decisiones de arquitectura de alto nivel vistas hasta ahora, generar diagramas de secuencia que representen la colaboración entre objetos del sistema.

El objetivo es ejercitar cómo se distribuyen las responsabilidades entre clases y componentes, y cómo se traduce el comportamiento dinámico del sistema.

2.A- Repaso teórico

- ¿Qué información brinda un diagrama de secuencia?

El Diagrama de Secuencia de UML es una herramienta que representa el **comportamiento dinámico de los objetos** dentro de un segmento de ejecución del software, mostrando el **flujo temporal de mensajes y llamadas** entre las instancias de las clases para entender su colaboración en una funcionalidad específica. Este diagrama se encuentra estrechamente vinculado a los casos de uso, proporcionando una visión detallada de cómo estos se implementan.

- ¿Qué elementos lo componen y qué significa cada uno de ellos?

El Diagrama de Secuencia de UML se utiliza para representar el comportamiento dinámico de los objetos en un segmento específico de ejecución del software. Sus componentes clave son las **Instancias de Clases**, ubicadas en la parte superior, que representan los objetos que interactúan; las **Líneas de Vida** (*lifelines*), que son las líneas verticales que indican la existencia de un objeto a lo largo del tiempo; y los **Mensajes**, representados por flechas entre las líneas de vida, que detallan la secuencia de llamadas, devoluciones de valores o señales que se envían entre los objetos siguiendo el flujo temporal descendente.

- ¿Qué relación existe entre los diagramas de secuencia y los casos de uso?

La relación principal es que el **diagrama de secuencia** está **estrechamente vinculado a los casos de uso**. Su propósito es proporcionar una **visión detallada de la implementación** de un caso de uso específico, representando el comportamiento de los objetos y la secuencia de mensajes necesarios para llevar a cabo la funcionalidad descrita por dicho caso de uso.

- ¿Se debe realizar un diagrama de secuencia por cada caso de uso?

No es obligatorio realizar un diagrama de secuencia por cada caso de uso. Por lo general, se elaboran para los casos de uso más relevantes o complejos, ya que su objetivo es profundizar en la lógica interna de las interacciones. También puede hacerse más de un diagrama para un mismo caso de uso si se desean representar escenarios alternativos o excepcionales.

○ ¿Qué diferencia hay entre un diagrama de secuencia de instancia y uno genérico?

La diferencia entre un diagrama de secuencia de instancia y uno genérico radica en el alcance de los escenarios que representan. El diagrama de instancia describe un escenario específico, normalmente el más común o exitoso de un caso de uso. En cambio, el diagrama genérico incluye todos los escenarios posibles, tanto el principal como los alternativos o de error, ofreciendo una visión más completa del comportamiento del sistema.

○ ¿De qué manera se relacionan los diagramas de secuencia y la arquitectura?

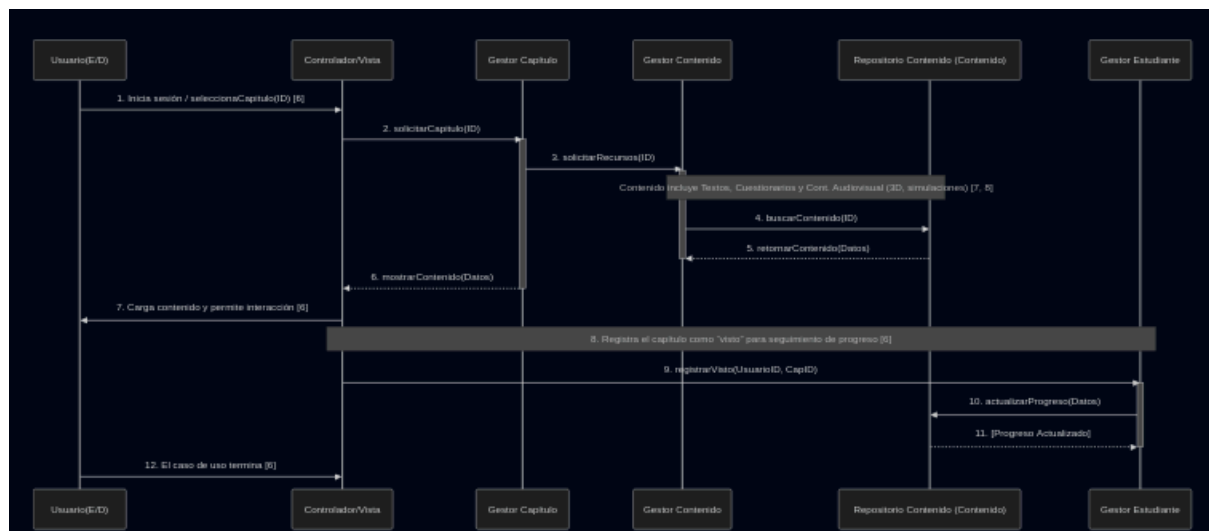
Los diagramas de secuencia se relacionan con la arquitectura porque reflejan cómo interactúan los distintos componentes definidos en el diseño del sistema. Permiten visualizar el flujo de mensajes entre capas, módulos o servicios, y sirven para validar decisiones arquitectónicas, comprobar la correcta asignación de responsabilidades y detectar posibles dependencias o acoplamientos entre partes del sistema.

2.B Selección de casos de uso

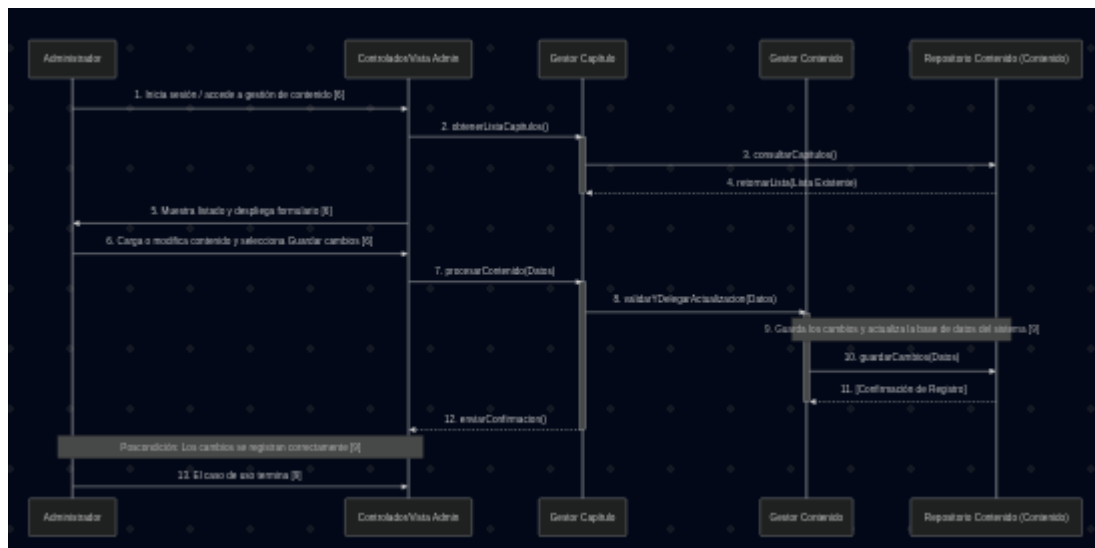
Seleccionamos como casos de usos más relevantes nuestro caso de uso 01 y 02, ya que los otros casos si bien son útiles sin estos no existiría el libro si quiera. Los demás agregan funciones y situaciones a estos:

- Caso_01: visualizar contenido
- Caso_02: administrar contenido

2.C Diagramación



[Imagen_1](#)



[Imagen_2](#)

2.D Análisis Final

Los diagramas de secuencia permiten visualizar de manera clara cómo se distribuyen las responsabilidades entre los distintos componentes definidos en la arquitectura del sistema. En el caso del **Libro Virtual Interactivo de Biología**, cada interacción entre los objetos muestra el recorrido de las solicitudes a través de las capas que forman parte del diseño arquitectónico: desde la interfaz o controlador/vista, pasando por los gestores (que representan la capa de aplicación o lógica de negocio), hasta llegar a los repositorios, donde se realiza el acceso a los datos.

Gracias a estos diagramas, se puede observar cómo el **controlador o vista** actúa como punto de entrada para los actores del sistema (usuario o administrador), cómo los **gestores** se encargan de coordinar las operaciones específicas —por ejemplo, recuperar contenido, registrar progreso o actualizar capítulos— y cómo los **repositorios** se encargan exclusivamente de la persistencia y consulta de información. Esta separación de responsabilidades refleja la estructura en capas planteada en la arquitectura, garantizando un bajo acoplamiento y una alta cohesión entre los componentes.

En síntesis, los diagramas de secuencia ayudan a comprender no solo el flujo de interacción entre los objetos, sino también cómo las decisiones de diseño arquitectónico (controladores, servicios, repositorios) se traducen en colaboraciones concretas dentro del sistema. Esto permite validar que la implementación respete la división de responsabilidades establecida y que cada capa cumpla su función de manera independiente pero coordinada dentro de la arquitectura general.