



ANÁLISIS Y DISEÑO DE SOFTWARE

# CONSTRUCCIÓN E INTERPRETACIÓN DE DIAGRAMAS DE SECUENCIA

# CONSTRUCCIÓN E INTERPRETACIÓN DE DIAGRAMAS DE SECUENCIA

## 1. Identificación del escenario

La construcción inicia con la identificación de un escenario funcional del sistema, comúnmente derivado de un caso de uso. Este escenario representa una tarea específica que el sistema debe realizar, como "registrar un usuario", "realizar una compra" o "consultar saldo".

**Ejemplo.** Se toma el caso de uso "realizar pago en línea".

## 2. Determinación de participantes

Se identifican los objetos o actores involucrados en el proceso. Cada uno será representado como una línea de vida (lifeline) en el diagrama. Los nombres de los participantes deben estar claros y reflejar su rol.

**Ejemplo.** Participantes: cliente:Usuario, sistema:CarritoCompras, banco:PasarelaPago.

## 3. Definición de los mensajes

Se establece el orden en que los participantes se comunican. Cada mensaje se representa mediante una flecha dirigida y debe llevar el nombre de la operación o método invocado. Si el mensaje espera una respuesta, se incluye un mensaje de retorno.

**Ejemplo.** El cliente envía el método `iniciarPago()` al sistema; este a su vez invoca `verificarTarjeta()` en el banco.

## 4. Representación de activaciones

Cada operación se ejecuta en un lapso temporal representado por una barra de activación en la línea de vida del objeto. Esta barra indica que el objeto está activo y procesando un mensaje.

## 5. Inclusión de estructuras de control

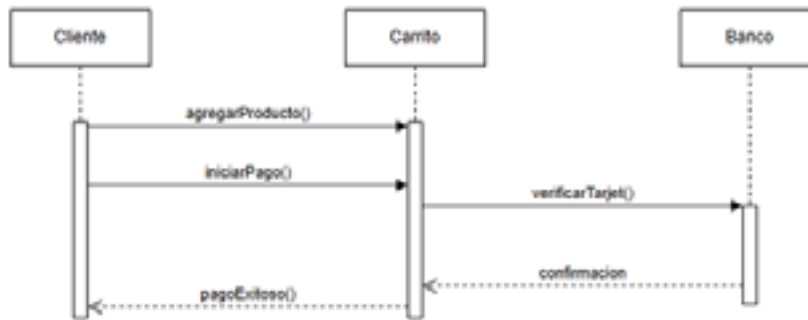
Si el flujo de mensajes implica condiciones, repeticiones o alternativas, se utilizan marcos de interacción (alt, loop, opt) para representar dichos comportamientos.

**Ejemplo.** En un proceso de pago, si la tarjeta es inválida, se incluye una rama alternativa donde el sistema informa del error al usuario (alt).

## 6. Revisión y validación

Una vez construido, el diagrama se revisa para asegurar que el orden de mensajes sea lógico y que todos los participantes estén correctamente representados. Esta validación permite identificar errores de lógica o de diseño antes de la implementación.

**Ejemplo práctico.** "Realizar una compra"



En este ejemplo, el cliente interactúa con el sistema para añadir un producto al carrito, luego solicita realizar el pago. El sistema valida la tarjeta con el banco y, si todo es correcto, responde con una confirmación de pago exitoso.

### Interpretación del diagrama

Interpretar un diagrama de secuencia implica seguir el flujo de mensajes desde arriba hacia abajo. El lector puede observar:

- El orden exacto de ejecución de los métodos.
- La relación entre los participantes.
- Las respuestas ante diferentes condiciones.

Esto permite al equipo de desarrollo validar requisitos, planificar la implementación y comunicar claramente el funcionamiento del sistema a todas las partes involucradas.