.UBAfiuba
FACULTAD DE INGENIERÍA

Programación orientada a eventos

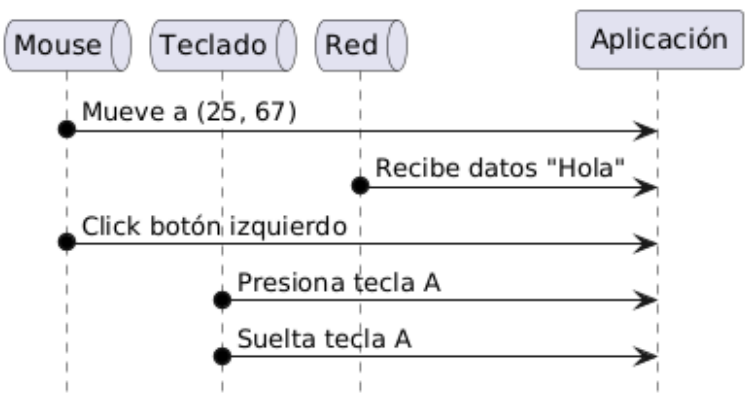
TB025 - Paradigmas de Programación

Curso Essaya

Programación orientada a eventos

Es un paradigma de programación en el que el **flujo del programa** es determinado por la ocurrencia de **eventos**, que son previstos pero no planeados (es decir, no se sabe cuándo ocurrirán).

Un ejemplo común es el desarrollo de **interfaces gráficas de usuario (GUI)**. En este caso, se deben manejar eventos como clics de ratón, pulsaciones de teclado, etc. Pero no es el único contexto en el que encuentra aplicación este paradigma.

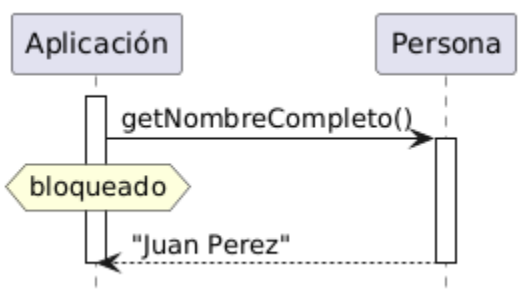


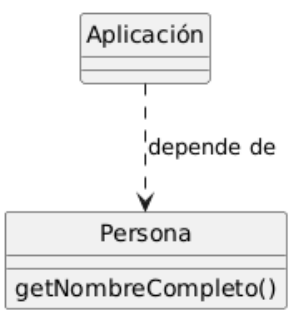
Fuente

Mensajes sincrónicos

Tradicionalmente los objetos se comunican mediante el envío **mensajes**, que en la mayoría de los lenguajes de programación se traduce directamente en la invocación de un **método**. De esta manera:

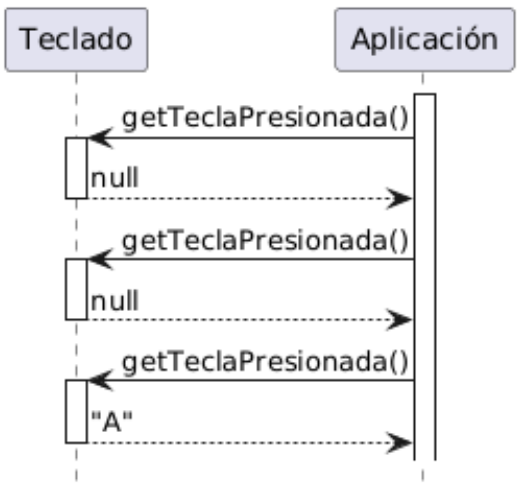
- La comunicación es **uno a uno** (un remitente, un destinatario).
- El receptor puede responder el mensaje en forma **sincrónica** (el remitente queda bloqueado hasta que el destinatario responde).
- El emisor depende del destinatario (ya que conoce su interfaz).





Eventos y *polling*

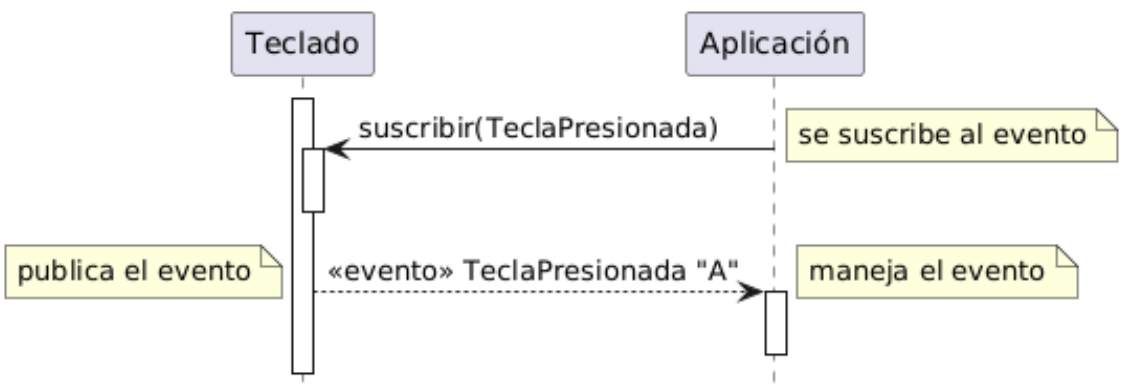
Una estrategia posible para procesar eventos es la de **polling** (encuestar), que suele ser ineficiente.

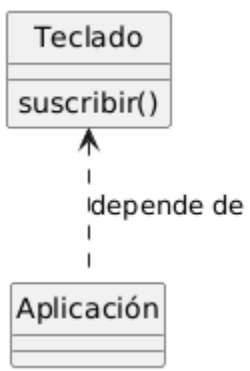


Publish/Subscribe

Una estrategia más eficiente es la de **publicación** y **suscripción** a eventos. De esta manera:

- La comunicación puede ser **uno a muchos** (un publicador, múltiples suscriptores).
- El suscriptor no responde al evento, sino que lo **maneja** de manera **asíncrona** (el publicador no queda bloqueado, sino que continúa ejecutando su código).
- La dependencia se invierte: el publicador no conoce a los suscriptores.

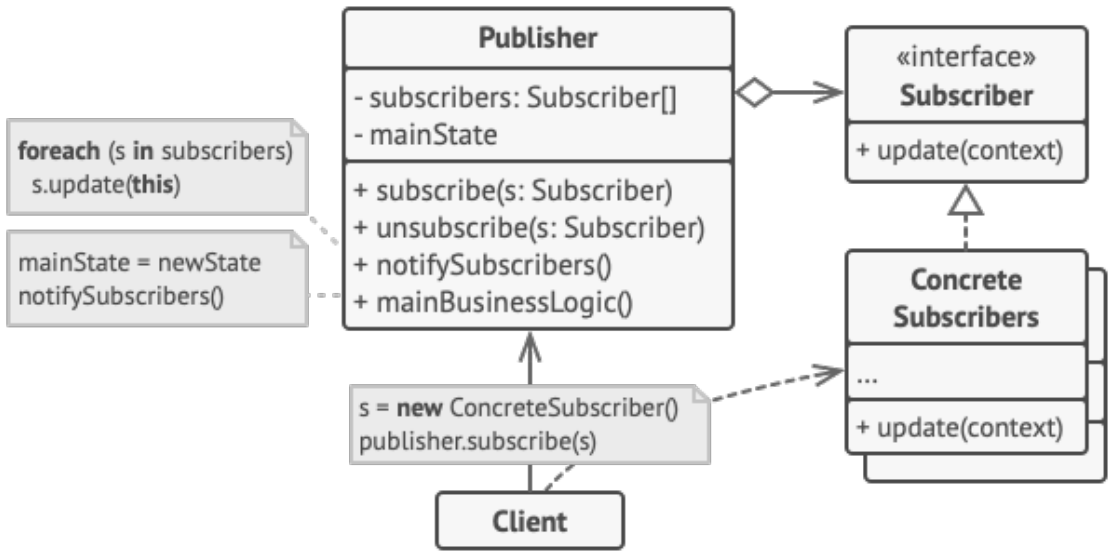




Fuente

Patrón Observer

Es un patrón de diseño de POO que implementa la estrategia de **Publish/Subscribe**.



Fuente