**Busca ejemplos de sistemas operativos según su clasificación y explica por qué los ejemplos encontrados se pueden encuadrar dentro de la clasificación elegida:**

**Por los servicios ofrecidos.**

**• Según el número de usuarios:**

o Monousuario

**Ejemplo:** **MS-DOS**  
**Explicación:** Solo permite que un usuario utilice el sistema a la vez. No admite sesiones simultáneas ni manejo de permisos para varios usuarios.

o Multiusuario

**Ejemplo:** **Linux (Kali, Ubuntu, Debian)** o **UNIX**  
**Explicación:** Varios usuarios pueden conectarse simultáneamente (por consola o remotamente), cada uno con su propio entorno y permisos. El sistema gestiona los recursos para evitar conflictos.

**• Según el número de tareas o procesos**

o Monotarea

**Ejemplo:** **MS-DOS**  
**Explicación:** Solo puede ejecutar un programa a la vez. Si el usuario quiere correr otro programa, debe cerrar el anterior.

o Multitarea

▪ Pseudomultitarea

**Ejemplo:** **Windows 3.1**  
**Explicación:** Alterna entre tareas muy rápido, pero solo ejecuta una realmente a la vez (no hay paralelismo real).

▪ Multitarea real

**Ejemplo:** **Linux**, **Windows 10**, **macOS**  
**Explicación:** Permite ejecutar varios procesos al mismo tiempo, gestionando la CPU entre ellos mediante planificación de procesos.

**• Según el número de procesadores del sistema informático.**

o Monoprocesador

**Ejemplo:** **MS-DOS**, **Windows 95**  
**Explicación:** Diseñados para equipos con un solo procesador. El sistema operativo no aprovecha múltiples núcleos.

o Multiprocesador

**Ejemplo:** **Linux**, **Windows 11 Server**, **macOS**  
**Explicación:** Soportan arquitecturas con varios procesadores o núcleos, distribuyendo las tareas entre ellos para mejorar el rendimiento.

**Por el tiempo de respuesta.**

• Proceso en lotes

**Ejemplo:** **OS/360 (IBM)** o **MS-DOS con scripts batch**  
**Explicación:** Ejecuta trabajos sin interacción del usuario. Las tareas se agrupan y se procesan una tras otra.

• Procesos en tiempo real

**Ejemplo:** **QNX**, **VxWorks**, **RTLinux**  
**Explicación:** Responden en un tiempo garantizado. Se usan en sistemas críticos (aeronáutica, automoción, robótica).

• Tiempo compartido

**Ejemplo:** **UNIX**, **Linux**, **Windows Server**  
**Explicación:** Varios usuarios acceden simultáneamente. El sistema reparte el tiempo del procesador para dar la impresión de ejecución simultánea.

**Por la forma de ofrecer los servicios.**

• Sistemas centralizados (o sistemas compartidos)

**Ejemplo:** **Mainframes con z/OS (IBM)**  
**Explicación:** Todos los recursos están en un solo sistema central y los usuarios acceden desde terminales.

• Sistemas distribuidos.

**Ejemplo:** **Google Android (basado en Linux)** o **Amoeba OS**  
**Explicación:** Varias computadoras trabajan coordinadamente, compartiendo tareas y recursos como si fueran un solo sistema.

• Sistemas operativos en red.

**Ejemplo:** **Windows Server**, **Novell NetWare**, **Linux con Samba/NFS**  
**Explicación:** Permiten a varias computadoras conectarse, compartir archivos, impresoras y otros recursos a través de una red.

**Por su disponibilidad.**

• Propietarios

**Ejemplo:** **Windows 11**, **macOS**, **z/OS**  
**Explicación:** El código fuente no está disponible al público. Su uso y modificación están restringidos por licencia.

• Libres

**Ejemplo:** **Kali Linux**, **Ubuntu**, **Debian**, **FreeBSD**  
**Explicación:** Su código fuente está disponible, se pueden modificar y distribuir libremente conforme a licencias como la GPL o BSD.