**ReactOS**

Es un sistema operativo de [código abierto](https://es.wikipedia.org/wiki/Código_abierto)está principalmente escrito en C, con algunos elementos, tales como ReactOS Explorer y la pila del sonido, escritos en C++. El proyecto se basa en MiniGW (*Minimalist GNU for Windows*). Los desarrolladores pretenden que las API del kernel y del modo de usuario sean más compatibles con Windows NT, en general también pretenden que el entorno grafico sea muy similar a Windows.

**Administración y planificación de procesos**

El planificador de procesos en ReactOS es un planificador de hilos que utiliza un diseño multitarea preventivo altamente escalable que por lo general lanza un nuevo hilo cada que se genera una interrupción al sistema. Los subprocesos tienen tres estados básicos: Ejecutar, Listo y programado. Los subprocesos pueden ser reemplazados por un subproceso de mayor prioridad, o interrumpidos por un DPC (llamada de procedimiento diferido) o APC (Advanced process control). Debido a esto, hay disponibles una variedad de funciones de bloqueo, sincronización, nivel de prioridad y enmascaramiento de interrupciones como:

• Evento (Event): una acción que generalmente se inicia fuera del alcance de un programa.  
• Semáforo (Semaphore): el método clásico para restringir el acceso a los recursos compartidos.  
• Temporizador (Timer): contadores que aumentan o disminuyen a una frecuencia fija.  
• Mutex (Mutex): un objeto que permite que varios subprocesos sincronicen el acceso y eviten el uso simultáneo de un recurso común.

También existe algo llamado Administrador de objetos el cual es un subsistema por el que deben pasar todos los demás subsistemas “ejecutivos” (que se encuentran contenidos en ntoskrnl.exe), para tener acceso a los recursos de ReactOS. Este estilo operativo garantiza que el Administrador de objetos se encargue de la administración de recursos para todos los demás subsistemas “ejecutivos” y así lograr asegurar que no existan tareas replicadas.

También encontramos el Controlador de caché, el cual proporciona memoria caché común para la I / O de archivos, al sincronizar con el Administrador de memoria, el Administrador de I / O y los controladores de I / O.

**Administrador de memoria**

ReactOS actualmente utiliza un Administrador de memoria que intenta imitar el comportamiento de Windows Server 2003. El Administrador de memoria es parte de ntoskrnl.exe. El cual contiene sub sistemas que se encargan de manejo de objetos, seguridad y gestión de procesos.

El Administrador de memoria maneja la memoria virtual, la protección de la memoria y la paginación de la memoria que pasa dentro y fuera de la memoria física al almacenamiento secundario, y aplica un asignado general de memoria física.

**Sistema de archivos**

Actual mente ReactOS solo soporta Sistema de archivos FAT, hay planes para que en un futuro soporte otros sistemas como NTFS, ext2.

**Comentario**

Me parece que aún tiene mucho trabajo por delante ya que en el tiempo que estuve probándolo note varios problemas como por ejemplo al crear una carpeta o eliminarla dentro del entorno grafico no se actualiza el icono hasta que vuelves a entrar al directorio, lo cual en un principio me generó confusión pues creí que no se estaba ejecutando el comando dado. Al instalar CodeBlocks no tuve problema alguno pero a la hora de intentar compilar un pequeño programa en C el sistema se congelo por completo y tuve que regresar a una versión anterior en la máquina virtual. También instale libre office el cual hasta donde pude observar no presentó ninguna falla en su ejecución. El CMD es muy similar al de Windows (incluso muchos de los comandos tienen nombres iguales). No sé si es muy limitado o falta más documentación pues hay algunos comandos que no encontré (como alguno que me permitiera ver los procesos que ese encuentren activos), en general los comandos que utilicé funcionaron cómo esperaba que lo hicieran. Pude crear directorios y archivos, ver sus contenidos borrarlos, moverlos, cambiarles el nombre etc…

Me parece que va por buen camino pues es innegable el parecido del entorno grafico al de un Windows viejo, si bien es cierto que todavía tiene deficiencias, la experiencia utilizándolo no fue mala considerando que está en estado alfa.

Me gusta la idea de un SO libre que sea compatible con Windows ya que muchos programas solo funcionan o están pensados para esta plataforma y a decir verdad no es la que más me agrada pero reconozco que tiene software que me gusta usar y que no es posible usar adecuadamente en otra plataforma.

**Fuentes**

<https://reactos.org/wiki/Kernel>

https://reactos.org/wiki/File\_Systems

<https://reactos.org/wiki/index.php?search=Memory+Manager&title=Special%3ASearch&profile=default&fulltext=1>

https://doxygen.reactos.org/dd/d06/ntoskrnl\_2mm\_2pagefile\_8c.html

**Pruebas de instalación y uso**

Instale el sistema en una máquina virtual de Oracle VM











