Taller de S.O.

Sexto Semestre ISC

JOSE ARTURO BUSTAMANTE LAZCANO

Contenido

[Introducción 3](#_Toc158633050)

[Definición 3](#_Toc158633051)

[Funciones Principales 3](#_Toc158633052)

[Ejemplos de Sistemas Operativos: 4](#_Toc158633053)

[Evolución Histórica: 4](#_Toc158633054)

[Primera Practica Instalando Instalación de Linux en Windows con WSL 5](#_Toc158633055)

[Cambio de la distribución predeterminada de Linux instalada 6](#_Toc158633056)

[Actualizar SubSistema Ubuntu 7](#_Toc158633057)

[Instalación de Subsistema Android 8](#_Toc158633058)

[Windows Subsystem for Android™️ 8](#_Toc158633059)

[GitHub para desarrolladores 8](#_Toc158633060)

[Programa de vista previa 9](#_Toc158633061)

[Configurado su entorno de desarrollo 9](#_Toc158633062)

[Requisitos previos 9](#_Toc158633063)

[Instalación de Amazon Appstore 9](#_Toc158633064)

[Configuración del Subsistema de Windows para Android™️ 10](#_Toc158633065)

[Pruebas y depuración 11](#_Toc158633066)

[Habilitación del modo de desarrollador en Configuración de Windows 11](#_Toc158633067)

[Conectarse al Subsistema de Windows para Android™️ para la depuración 11](#_Toc158633068)

[Conexión a un dispositivo de prueba 12](#_Toc158633069)

[Depurar la aplicación 12](#_Toc158633070)

[Creación de APK universales 13](#_Toc158633071)

[Consideraciones sobre la compatibilidad de entrada para dispositivos Windows 13](#_Toc158633072)

[Entrada de teclado 14](#_Toc158633073)

[Entrada de mouse 14](#_Toc158633074)

[Administración y redimensionamiento de ventanas 15](#_Toc158633075)

[Requisito mínimo de pantalla 15](#_Toc158633076)

[Formatos letterbox y pillarbox 15](#_Toc158633077)

[Consideraciones adicionales sobre el redimensionamiento 16](#_Toc158633078)

[Eventos del ciclo de vida de aplicaciones 16](#_Toc158633079)

[Consideraciones sobre el ciclo de vida de las VM 17](#_Toc158633080)

[Propiedades de máquina virtual 17](#_Toc158633081)

[Redireccionamiento a aplicaciones de Windows 18](#_Toc158633082)

[Seguridad 19](#_Toc158633083)

[Desinstalación del subsistema de Windows para Android™️ 19](#_Toc158633084)

[Solución de problemas 19](#_Toc158633085)

[Recursos adicionales 20](#_Toc158633086)

# Introducción

Un sistema operativo (SO) es un software fundamental que gestiona los recursos de hardware y proporciona servicios a los programas de aplicación. Actúa como intermediario entre el hardware y los programas de software, facilitando la comunicación y la interacción entre ellos. Aquí tienes una introducción básica a los sistemas operativos:

Definición:

Un sistema operativo es un conjunto de programas que gestionan los recursos de hardware de un sistema informático y proporcionan servicios a los programas de aplicación.

Funciones Principales:

Gestión de Recursos:

Procesador: Asigna tiempo de CPU a los diferentes procesos.

Memoria: Controla la asignación y liberación de memoria.

Dispositivos de Entrada/Salida: Coordina las operaciones de E/S.

Interfaz de Usuario:

Interfaz de línea de comandos (CLI) y gráfica (GUI): Permite la interacción entre el usuario y el sistema.

Gestión de Archivos:

Sistemas de archivos: Organiza y almacena datos en dispositivos de almacenamiento.

Seguridad:

Control de Acceso: Protege recursos limitando el acceso a usuarios autorizados.

Gestión de Contraseñas: Asegura la autenticación de usuarios.

Gestión de Procesos:

Planificación de Procesos: Decide qué proceso se ejecuta en un momento dado.

Comunicación entre Procesos: Facilita la comunicación y sincronización entre procesos.

Red:

Gestión de Red: Facilita la conexión y la comunicación en redes de computadoras.

Tipos de Sistemas Operativos:

Monotarea:

Ejecuta un solo programa a la vez.

Multitarea:

Ejecuta múltiples tareas simultáneamente.

Monousuario:

Diseñado para un solo usuario.

Multiusuario:

Permite que múltiples usuarios utilicen el sistema simultáneamente.

## Ejemplos de Sistemas Operativos:

Windows: Desarrollado por Microsoft, común en entornos de escritorio.

Linux: Sistema operativo de código abierto utilizado en servidores y sistemas integrados.

macOS: Sistema operativo de Apple, diseñado para hardware Mac.

Unix: Base para muchos sistemas operativos, utilizado en entornos empresariales.

## Evolución Histórica:

Década de 1950: Sistemas operativos simples para mainframes.

Década de 1960: Desarrollo de sistemas multiprogramación.

Década de 1970: Surgimiento de sistemas operativos como UNIX.

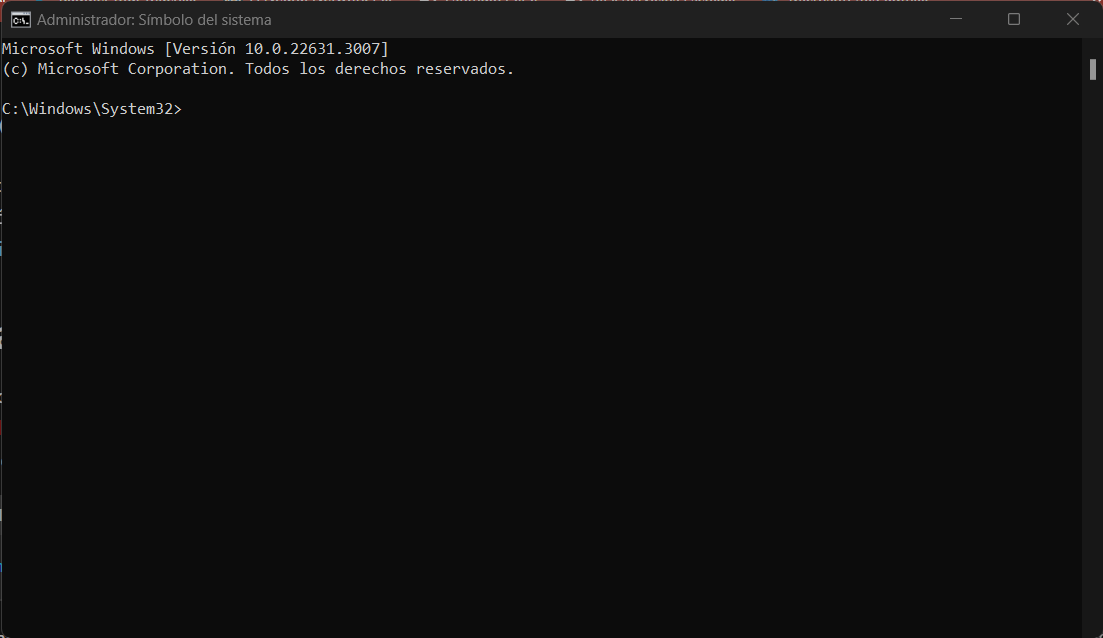
Década de 1980: Popularización de sistemas operativos de escritorio como MS-DOS y Windows.

Década de 1990 en adelante: Auge de sistemas operativos de código abierto como Linux.

# Primera Practica Instalando Instalación de Linux en Windows con WSL

<https://learn.microsoft.com/es-es/windows/wsl/install>

Paso de Instalación



Comando de instalación de WSL

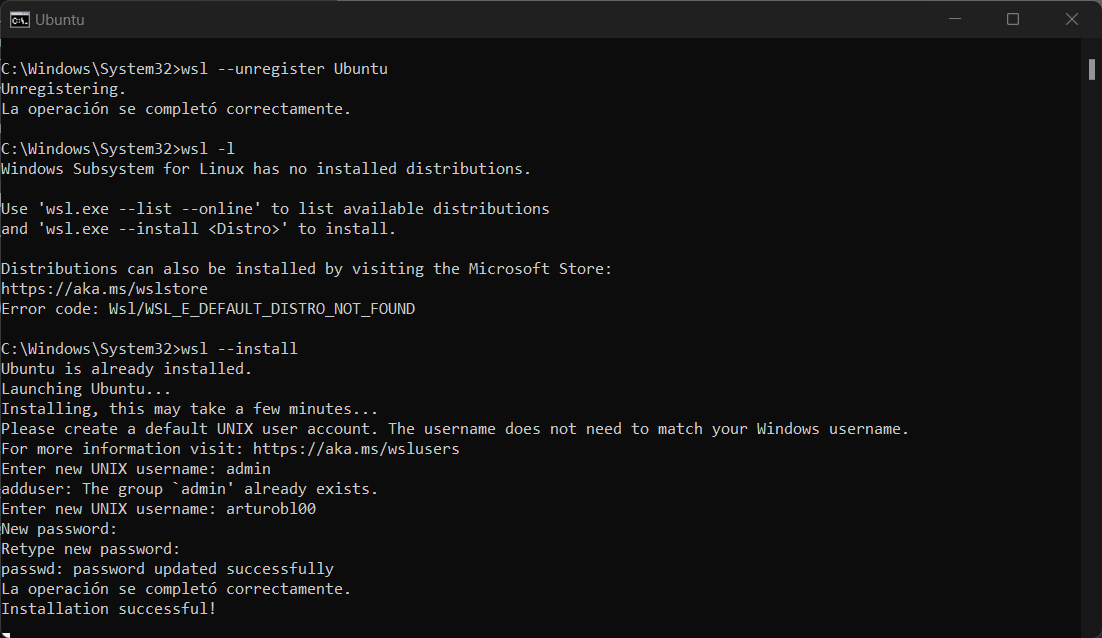
Ahora puede instalar todo lo que necesita para ejecutar WSL con un solo comando. Abra PowerShell o el símbolo del sistema de Windows como administrador; para ello, haga clic con el botón derecho y seleccione "Ejecutar como administrador", escriba el comando wsl --install y reinicie la máquina.

PowerShell

Copiar

wsl --install

Este comando habilitará las características necesarias para ejecutar WSL e instalará la distribución Ubuntu de Linux. (Esta distribución predeterminada se puede cambiar).



## Cambio de la distribución predeterminada de Linux instalada

De forma predeterminada, la distribución de Linux instalada será Ubuntu. Se puede cambiar mediante la marca -d.

* Para cambiar la distribución instalada, escriba: wsl --install -d <Distribution Name>. Reemplace <Distribution Name> por el nombre de la distribución que desea instalar.
* Para ver una lista de las distribuciones de Linux disponibles para descargar a través de la tienda en línea, escriba wsl --list --online o wsl -l -o.
* Para instalar distribuciones de Linux adicionales después de la instalación inicial, también puede usar el comando wsl --install -d <Distribution Name>.

**Sugerencia**

Si desea instalar distribuciones adicionales desde dentro de una línea de comandos de Linux/Bash (en lugar de hacerlo desde PowerShell o el símbolo del sistema), debe usar .exe en el comando wsl.exe --install -d <Distribution Name> o para enumerar las distribuciones disponibles: wsl.exe -l -o.

Si experimenta un problema durante el proceso de instalación, consulte la [sección de instalación de la guía de solución de problemas](https://learn.microsoft.com/es-es/windows/wsl/troubleshooting#installation-issues).

Para instalar una distribución de Linux que no aparece como disponible, puede [importar cualquier distribución de Linux](https://learn.microsoft.com/es-es/windows/wsl/use-custom-distro) mediante un archivo TAR. O bien, en algunos casos, [como con Arch Linux](https://wsldl-pg.github.io/ArchW-docs/How-to-Setup/), puede instalarla mediante un archivo .appx. También puede crear su propia [distribución personalizada de Linux](https://learn.microsoft.com/es-es/windows/wsl/build-custom-distro) para usarla con WSL.

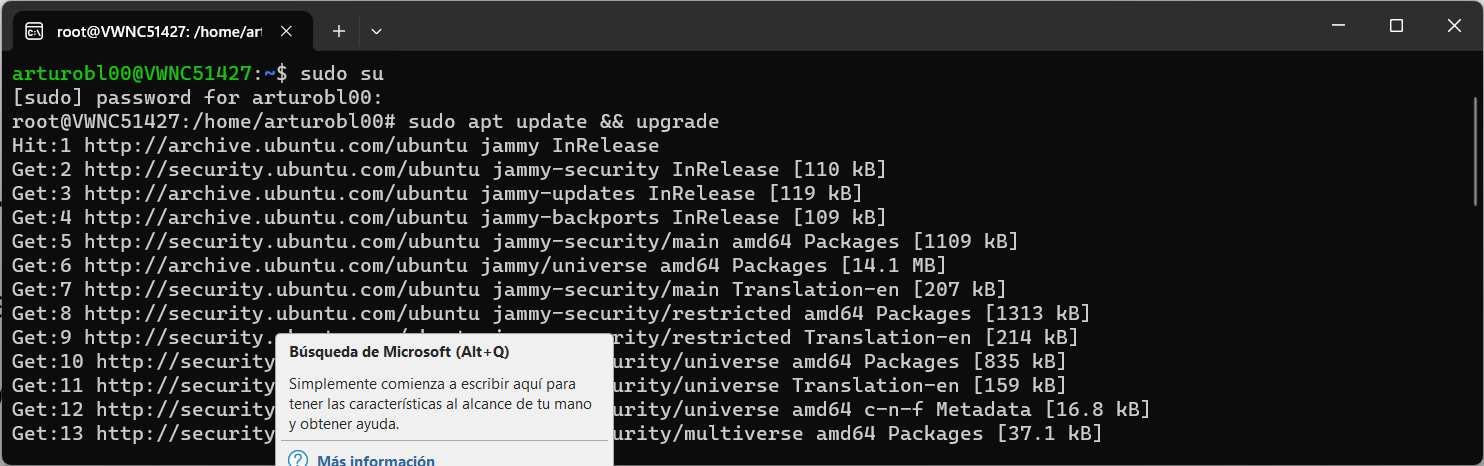
## Actualizar SubSistema Ubuntu

Cambiamos a super usuario

arturobl00@VWNC51427:~$ sudo su

Ejecutamos actualización

arthurbl@VWNC51427:~$ sudo apt update && upgrade



Instalación de entorno grafico

sudo apt install xfce4

Ejecutar Entorno

sudo startxfce4

# Instalación de Subsistema Android

# Windows Subsystem for Android™️

Windows Subsystem for Android™ permite que un dispositivo con Windows 11 ejecute aplicaciones de Android que están disponibles en Amazon Appstore. Android es una marca comercial de Google LLC. Si es desarrollador y le interesa apuntar a dispositivos de escritorio Windows y optimizar para el sistema operativo Windows, esta guía es para usted.

Para que la aplicación Android esté disponible en dispositivos Windows 11, debe:

* [Enviar la aplicación a Amazon AppStore](https://developer.amazon.com/apps-and-games/appstore-on-windows-11).

Para obtener más información o soporte técnico:

* [Regístrese para obtener actualizaciones en Amazon AppStore en el programa de Windows](https://m.amazonappservices.com/developer-interest).
* [Visite el portal de soporte técnico para desarrolladores de Amazon](https://developer.amazon.com/apps-and-games/appstore-on-windows-11), donde puede encontrar artículos, foros, preguntas más frecuentes o ponerse en contacto con el soporte técnico directo a través de la página "Póngase en contacto con nosotros" de Appstore una vez que haya configurado una cuenta de desarrollador de Amazon.

Esta guía puede ayudarle a probar y depurar la aplicación Android en Windows:

* [Configurar el entorno de desarrollo](https://learn.microsoft.com/es-es/windows/android/wsa/#set-up-your-development-environment), incluidos los [requisitos previos](https://learn.microsoft.com/es-es/windows/android/wsa/#prerequisites), la [instalación de Amazon Appstore](https://learn.microsoft.com/es-es/windows/android/wsa/#install-the-amazon-appstore) y el uso de la [Configuración](https://learn.microsoft.com/es-es/windows/android/wsa/#windows-subsystem-for-android-settings).
* [Probar y depurar](https://learn.microsoft.com/es-es/windows/android/wsa/#test-and-debug) una aplicación en un dispositivo con Windows 11.
* Controlar las consideraciones de compatibilidad de dispositivos Windows, por ejemplo: la [entrada de teclado](https://learn.microsoft.com/es-es/windows/android/wsa/#keyboard-input), la [entrada del mouse](https://learn.microsoft.com/es-es/windows/android/wsa/#mouse-input) y la [administración y redimensionamiento de ventanas](https://learn.microsoft.com/es-es/windows/android/wsa/#window-management-and-resizing).
* [Solucionar problemas y encontrar respuestas](https://learn.microsoft.com/es-es/windows/android/wsa/#troubleshooting-issues).

## GitHub para desarrolladores

¿Quiere obtener más información sobre el subsistema de Windows para Android™️ hoja de ruta, analice los problemas de desarrollador y los errores de archivo o las solicitudes de características con el equipo del subsistema? Visite el [Subsistema de Windows para Android™️ Desarrolladores de GitHub](https://github.com/microsoft/WSA).

## Programa de vista previa

La versión preliminar del programa Windows Subsystem for Android™️ permite a los usuarios recibir compilaciones preliminares anticipadas de Windows Subsystem for Android™️ y Amazon Appstore en Windows. Para más información, visite la [página de la versión preliminar del programa](https://learn.microsoft.com/es-es/windows/android/wsa/preview-program).

## Configurado su entorno de desarrollo

Para probar la aplicación de Android en el entorno de escritorio de Windows, se requiere cierta configuración.

### Requisitos previos

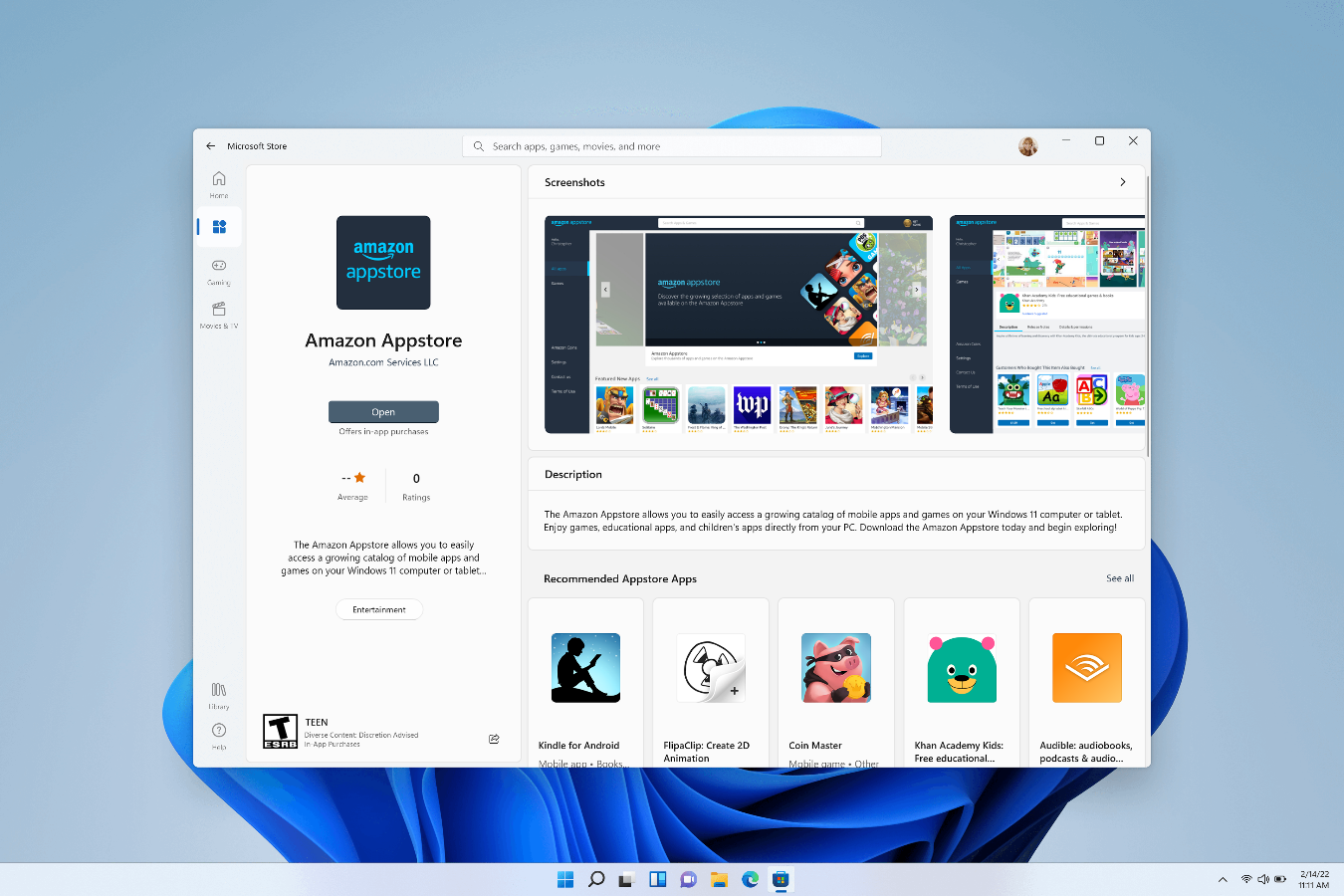
Subsistema de Windows para Android™️ está disponible en Windows 11. El dispositivo debe cumplir requisitos específicos: [Requisitos del dispositivo](https://support.microsoft.com/windows/f8d0abb5-44ad-47d8-b9fb-ad6b1459ff6c).

### Instalación de Amazon Appstore

Microsoft Store instalará automáticamente subsistema de Windows para Android™️ en segundo plano cuando se realicen cualquiera de las dos acciones de usuario siguientes:

1. Se instala [Amazon Appstore desde Microsoft Store](https://www.microsoft.com/store/productId/9NJHK44TTKSX). Al seleccionar **Obtener**, comenzará la instalación de la aplicación.
2. Se instala una aplicación de Android desde Microsoft Store por primera vez, lo que también instalará Amazon Appstore.

A continuación, la aplicación **Amazon Appstore** aparecerá en el menú Inicio de Windows 11 y estará disponible en la búsqueda, lo que ofrece un catálogo de aplicaciones de Android. La **subsistema de Windows para Android™️** aplicación, que permite controlar la configuración y las características de la aplicación móvil, también aparecerá en el menú Inicio.



**Nota**

Amazon AppStore en Windows (un requisito para ejecutar aplicaciones Android en Windows 11) está disponible en [**regiones seleccionadas**](https://support.microsoft.com/windows/countries-and-regions-that-support-amazon-appstore-on-windows-d8dd17c7-5994-4187-9527-ddb076f9493e?preview=true).

## Configuración del Subsistema de Windows para Android™️

Para modificar el Subsistema de Windows para Android™️ configuración, vaya a: **Iniciar > Todas las aplicaciones > Subsistema de Windows para Android™️**. Más información sobre las características específicas de la aplicación de configuración: [Administración de la configuración de aplicaciones móviles en Windows](https://support.microsoft.com/windows/000f97e8-8c20-490e-9ef4-cd90d903f847).



## Pruebas y depuración

Para probar y depurar la aplicación en un dispositivo Windows 11 mediante el Subsistema de Windows para Android️™se requieren los siguientes pasos de configuración.

### Habilitación del modo de desarrollador en Configuración de Windows

Primero tiene que habilitar el modo de desarrollador. Abra la configuración del [**Subsistema de Windows para Android™️**](https://learn.microsoft.com/es-es/windows/android/wsa/#windows-subsystem-for-android-settings). Una vez abierto, habilite el **modo desarrollador** en **Configuración avanzada**.

### Conectarse al Subsistema de Windows para Android™️ para la depuración

Para conectarse al subsistema de Windows para Android️™máquina virtual para la depuración:

1. Inicie una aplicación Android que se instaló con Amazon Appstore.
2. Puede conectarse mediante adb connect con el siguiente comando (debe tener [instalado adb](https://developer.android.com/studio/command-line/adb)):

PowerShellCopiar

adb connect 127.0.0.1:58526

### Conexión a un dispositivo de prueba

Para conectarse a un dispositivo de prueba (con subsistema de Windows para Android™️ instalado) en la misma red desde Windows/Mac:

1. En el dispositivo de prueba (donde está instalado el Subsistema de Windows para Android️™) abra una ventana de PowerShell e identifique la dirección IP del dispositivo de prueba ejecutando el comando :

PowerShellCopiar

ipconfig

1. Mediante el terminal del dispositivo de depuración donde Android Studio y Android SDK están instalados (Mac/Windows), escriba el comando:

ConsolaCopiar

adb connect <TEST DEVICE IP ADDRESS>:58526

<TEST DEVICE IP ADDRESS> se puede encontrar en la salida de "ipconfig" desde el dispositivo de prueba. También puede implementar y depurar aplicaciones desde Android Studio.

Para usar Android Debug Bridge (ADB) para conectar la estación de trabajo de desarrollo directamente al dispositivo Android para que pueda instalar paquetes y evaluar los cambios, consulte [Android Debug Bridge en la documentación del Proyecto de código abierto de Android](https://source.android.com/setup/build/adb).

### Depurar la aplicación

Aunque las aplicaciones deben instalarse mediante Amazon Appstore, la depuración de una aplicación Android en un dispositivo Windows es posible mediante un APK (paquete de aplicaciones Android) y adb (Android Debug Bridge).

Para depurar un APK mediante adb:

1. Siga los pasos para conectarse al subsistema de Windows para Android™️ máquina virtual anterior.
2. Instale el APK mediante el comando adb install: adb install app-debug.apk.

Salida esperada:

PowerShellCopiar

Performing Streamed Install

Success

1. Aparecerá una notificación con el mensaje "aplicación instalada" en el menú de notificaciones de Windows, y la aplicación se iniciará una vez que se seleccione.

### Creación de APK universales

Subsistema de Windows para Android™️ utiliza la tecnología Intel Bridge para habilitar aplicaciones Arm en procesadores basados en x86. Por supuesto, las aplicaciones de ARM se ejecutarán en procesadores basados en ARM de forma nativa. La capa de emulación provocará una sobrecarga de rendimiento: para obtener un rendimiento óptimo, envíe la aplicación para las arquitecturas x86-64 y ARM64.

## Consideraciones sobre la compatibilidad de entrada para dispositivos Windows

Hay algunos comportamientos de entrada únicos que se deben tener en cuenta y que probablemente exijan actualizaciones en el código de la aplicación de Android, diseñada para dispositivos portátiles, para que sea compatible cuando se ejecute en un dispositivo de escritorio Windows través de Amazon Appstore.

### Entrada de teclado

En el caso de los campos de entrada de texto que controla un método de entrada de teclado virtual en pantalla (o IME), como EditText, las aplicaciones deben comportarse según lo previsto. ([Clase EditText en la documentación de Android](https://developer.android.com/reference/android/widget/EditText)).

En el caso de las pulsaciones de teclas que el marco no pueda prever, las aplicaciones tendrán que controlar el comportamiento por sí mismas. Si esto ya se ha implementado en la aplicación, no se requiere ningún trabajo adicional.

Por ejemplo, es posible que algunos juegos ya admitan el movimiento facilitado mediante el teclado, a través de las teclas w, a, s, d, junto con la entrada táctil.

A continuación encontrará las entradas de teclado para las que los desarrolladores deben tener en cuenta las actualizaciones de código al compilar para dispositivos con Windows 11:

* Tecla ENTRAR
* Navegación con las teclas de flecha y la tecla de tabulación
* Cambiar el color resaltado del elemento seleccionado
* Métodos abreviados basados en CTRL

Consulte la documentación de Android para obtener más información sobre la optimización para estos escenarios de entrada del teclado en dispositivos de escritorio:

* [Guía de compatibilidad de entradas en la documentación de Android](https://developer.android.com/topic/arc/input-compatibility)
* [Guía de control de las entradas del teclado en la documentación de Android](https://developer.android.com/training/keyboard-input)
* [Guía de uso de gestos táctiles en la documentación de Android](https://developer.android.com/training/gestures)

### Entrada de mouse

Los desarrolladores deben considerar la posibilidad de actualizar el código de las siguientes entradas del mouse al compilar para dispositivos Windows:

* Clic con el botón derecho
* Información sobre herramientas/mantener el puntero sobre el texto
* Efectos de mantener el puntero
* Acción de la rueda de desplazamiento del mouse
* Arrastrar y colocar

La entrada del mouse, de manera similar a la entrada del teclado, debe seguir las directrices oficiales para aplicaciones de Android. Esto significa usar la clase InputDevice emparejada con la constante SOURCE\_MOUSE. Consulte la documentación de Android para obtener más información sobre la optimización para estos escenarios de entrada del mouse en dispositivos de escritorio:

* [Guía de compatibilidad de entradas en la documentación de Android](https://developer.android.com/topic/arc/input-compatibility)
* [Referencia de InputDevice en la documentación de Android](https://developer.android.com/reference/android/view/InputDevice)
* [Referencia de SOURCE\_MOUSE en la documentación de Android](https://developer.android.com/reference/android/view/InputDevice#SOURCE_MOUSE)

## Administración y redimensionamiento de ventanas

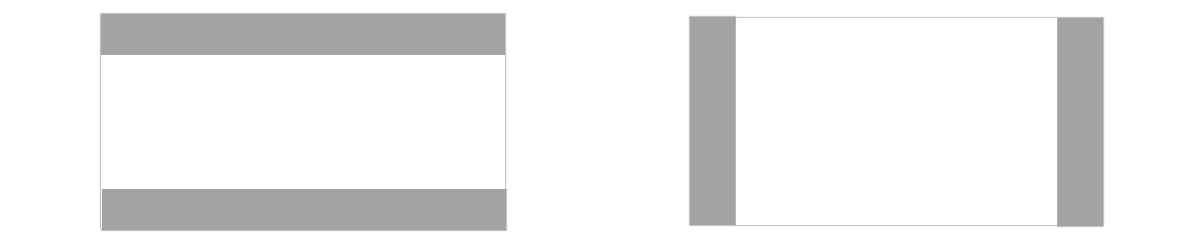
A diferencia de los factores de forma móviles tradicionales, las aplicaciones de Android que se ejecutan en Windows 11 se pueden redimensionar libremente, deben tener capacidad de respuesta en su cambio de tamaño, y se pueden ajustar mediante acciones o gestos de Windows.

### Requisito mínimo de pantalla

Windows 11 aplica un requisito mínimo de pantalla con una resolución de 720p (1280×720) con una pantalla de más de >9".

### Formatos letterbox y pillarbox

Cuando la relación de aspecto del tamaño de un ventana no se alinea entre los tamaños de pantalla del dispositivo en el que se muestra la ventana, el resultado puede ser el formato letterbox (la ventana es más ancha que alta, u horizontalmente más larga) o pillarbox (la ventana es más estrecha que el ancho, o verticalmente más larga). Como consecuencia, se colocan barras a los lados de la ventana para centrarla. Estas barras pueden ser de color claro u oscuro, en función de la configuración del sistema seleccionada. Esto solo tendrá lugar cuando la aplicación de Android se ajuste o maximice, lo que permite a las aplicaciones de Android aprovechar las características de ajuste enriquecidas en Windows e integrarlas en el modelo basado en ventanas.



### Consideraciones adicionales sobre el redimensionamiento

También se debe tener en cuenta lo siguiente al actualizar una aplicación de Android para que se ejecute en un dispositivo con Windows 11 con respecto a la administración y el redimensionamiento de ventanas:

* Tamaño de inicio original
* Dimensiones de la ventana
  + Límites del contenido
  + Tamaño de forma libre
* Orientación de la pantalla

Para obtener más información sobre la optimización para los escenarios de cambio de tamaño de las ventanas en dispositivos de escritorio, siga la [guía de administración de ventanas en la documentación de Android](https://developer.android.com/topic/arc/window-management).

## Eventos del ciclo de vida de aplicaciones

El desarrollo de aplicaciones Android en entornos de varias ventanas afecta a los eventos de ciclo del vida que decida usar en la aplicación. Aunque la invalidación del evento onPause puede lograr los resultados deseados en un teléfono o tableta, es normalmente el evento incorrecto que se usará si cambia la experiencia de usuario de la aplicación.

Consulte la [documentación de Android](https://developer.android.com/guide/components/activities/activity-lifecycle) para ver una descripción de los eventos del ciclo de vida. La mayoría de las veces, le interesará usar el evento onStop y no los eventos onPause o onUserLeaveHint. De hecho, muchas implementaciones de Android de varias ventanas no entregan la notificación de onUserLeaveHint y, por tanto, cualquier lógica crítica para la empresa que pueda estar en ese controlador de eventos no se llamará en estas plataformas, incluido el Subsistema de Windows para Android™️.

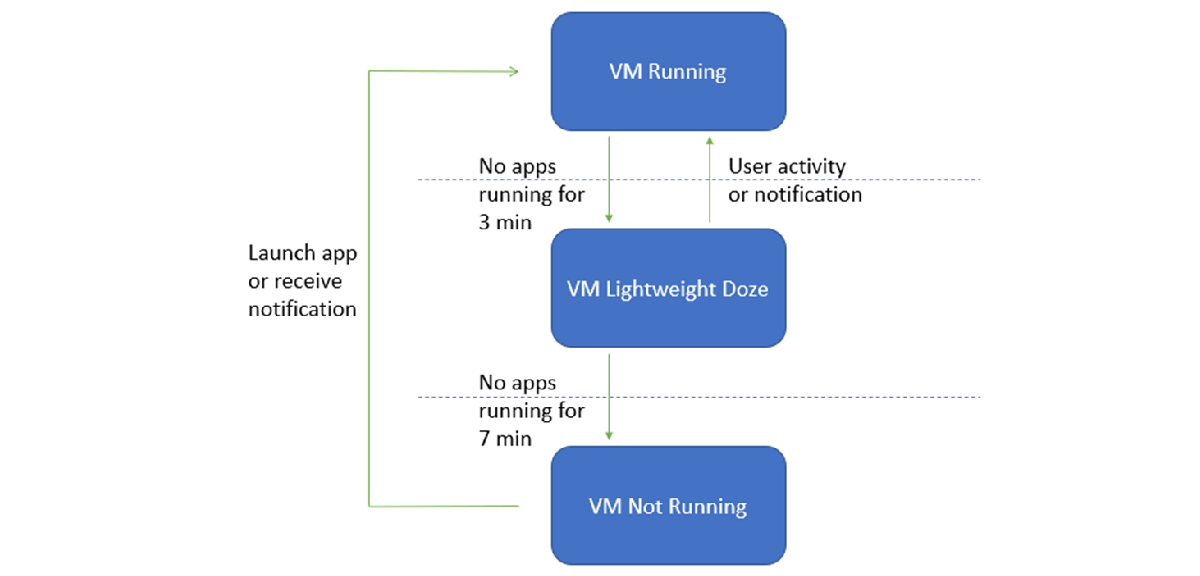
## Consideraciones sobre el ciclo de vida de las VM

Subsistema de Windows para Android™️ utiliza una máquina virtual (VM) que proporciona compatibilidad con el [marco de AOSP de](https://source.android.com/)y dispositivos como teclados, ratones, toques, lápiz, etc.

Hay tres estados posibles para la máquina virtual que ejecuta aplicaciones con subsistema de Windows para Android™️:

1. En ejecución
2. Lightweight Doze (Suspensión ligera): Se activa después de que no haya actividad de aplicación durante 3 minutos. Se desactiva con la actividad del usuario o una notificación de la aplicación.
3. No en ejecución: Se activa después de que no haya actividad de aplicación durante 7 minutos.

La actividad del usuario desencadena las transiciones entre estos estados, como el inicio o la interacción con la aplicación de Android o una notificación de la aplicación. Las aplicaciones de Android se pausan y luego se detienen cuando se minimiza su ventana.



### Propiedades de máquina virtual

Las propiedades del Subsistema Windows para Android™️ VM se enumeran a continuación. No se recomienda codificar estos valores de forma rígida, ya que podría provocar incompatibilidades futuras.

Expandir tabla

| **Propiedad** | **Value** |
| --- | --- |
| Build.MANUFACTURER | Microsoft Corporation |
| Build.MODEL | Subsystem for Android(TM) |
| Build.VERSION.SDK\_INT | 32 |
| Build.BOARD | windows |

## Redireccionamiento a aplicaciones de Windows

Subsistema de Windows para Android™️ redirige automáticamente las intenciones de archivos y esquemas URI comunes al controlador de protocolo o archivo predeterminado de Windows correspondiente (si coinciden varios filtros de intención, los usuarios ven una opción "Aplicación predeterminada de Windows" en el cuadro de diálogo del selector). Las intenciones de archivo admitidas incluyen [ACTION\_VIEW](https://developer.android.com/reference/android/content/Intent#ACTION_VIEW), [ACTION\_EDIT](https://developer.android.com/reference/android/content/Intent#ACTION_EDIT), [ACTION\_SEND](https://developer.android.com/reference/android/content/Intent#ACTION_SEND)y [ACTION\_SEND\_MULTIPLE](https://developer.android.com/reference/android/content/Intent#ACTION_SEND_MULTIPLE), que copian el archivo en la carpeta Descargas de Windows antes de abrirlo. Las intenciones de URI admitidas incluyen [ACTION\_VIEW](https://developer.android.com/reference/android/content/Intent#ACTION_VIEW) para los esquemas http/https y [ACTION\_VIEW](https://developer.android.com/reference/android/content/Intent#ACTION_VIEW) y [ACTION\_SENDTO](https://developer.android.com/reference/android/content/Intent#ACTION_SENDTO) para el esquema mailto.

Las aplicaciones android también pueden redirigir manualmente a las aplicaciones de Windows mediante esquemas de URI personalizados. Establezca la acción de intención en com.microsoft.windows.LAUNCH\_URI y agregue una cadena adicional a la intención denominada com.microsoft.windows.EXTRA\_URI con el URI personalizado como valor. Por ejemplo, para iniciar la aplicación Calculadora de Windows desde una aplicación Android (Java):

JavaCopiar

Intent intent = new Intent("com.microsoft.windows.LAUNCH\_URI");

intent.putExtra("com.microsoft.windows.EXTRA\_URI", "ms-calculator:");

try {

startActivity(intent);

} catch (ActivityNotFoundException e) {

// Not running in Windows Subsystem for Android™️ (or running on an older build that did not contain this feature).

}

## Seguridad

Tanto los controladores Windows en modo kernel como las aplicaciones de Windows que se ejecutan en el nivel de integridad (IL) medio pueden inspeccionar contenedores arbitrarios de Android y la memoria de las aplicaciones de Android. No hay ningún plan para agregar detección de comportamientos sospechosos, macro, bots o engaños a corto plazo.

Los desarrolladores que consulten getSecurityLevel obtendrán SECURITY\_LEVEL\_SW\_SECURE\_CRYPTO. Obtenga más información sobre getSecurityLevel en la [guía de referencia de API de Android](https://developer.android.com/reference/android/media/MediaDrm#getSecurityLevel(byte%5B%5D)).

## Desinstalación del subsistema de Windows para Android™️

Puede desinstalar el Subsistema de Windows para Android™️, pero tenga en cuenta que también se desinstalarán todas las aplicaciones asociadas.

* Al desinstalar Amazon Appstore, se desinstalará el Subsistema de Windows para Android™️ y todas las demás aplicaciones de Android.
* La desinstalación de una aplicación de Amazon Appstore solo desinstalará la aplicación (el mismo comportamiento que para las aplicaciones de Windows).
* Al desinstalar el Subsistema de Windows para Android️™se desinstalarán Amazon Appstore y todas las aplicaciones de Android.

## Solución de problemas

Si encuentra problemas específicos de Amazon Appstore en Windows, pruebe los siguientes pasos de solución de problemas:

1. Seleccione la **búsqueda de Windows** en la barra de tareas de Windows.
2. Busque "Amazon Appstore" y haga clic con el botón derecho en el icono de Amazon Appstore.
3. Seleccione "Aplicación Configuración" en las opciones de la lista desplegable.
4. Seleccione "Almacenamiento y caché" y haga clic en "Clear Storage" (Borrar almacenamiento) y "Borrar caché".
5. Vuelva y seleccione "Force Stop" (Forzar detención).
6. Cierre la ventana de Configuración de Amazon Appstore.
7. Vuelva a iniciar Amazon Appstore.

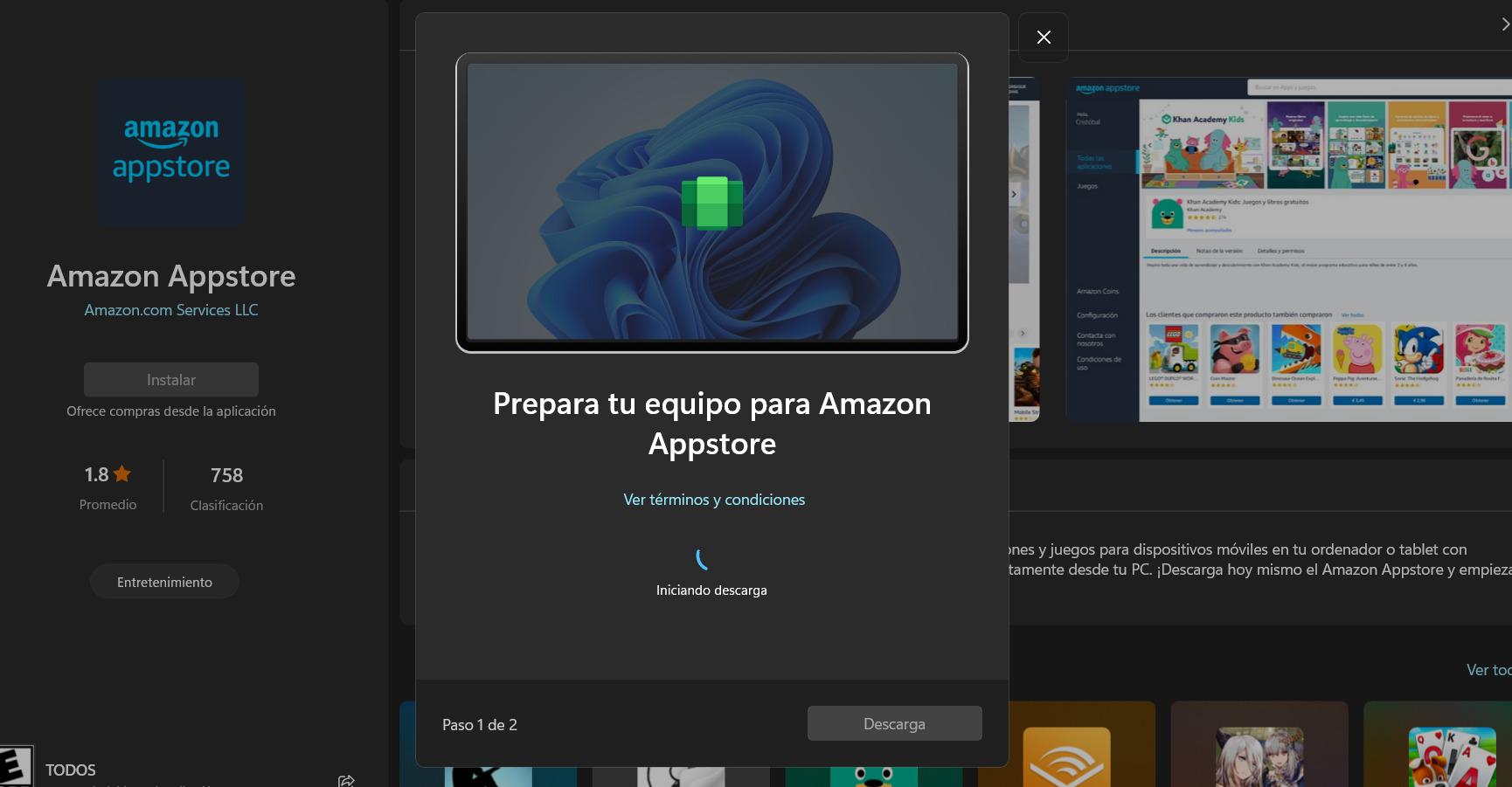
Para obtener más pasos de solución de problemas relacionados con la aplicación Subsistema de Windows para Android™️ Configuración o para dejar comentarios mediante el Centro de opiniones, vea [Solución de problemas y preguntas más frecuentes sobre aplicaciones móviles en Windows](https://support.microsoft.com/windows/d6062afd-98a1-4018-a7c8-6b3b680a2ea5).

Para cualquier otra pregunta y soporte técnico para desarrolladores, usa la [etiqueta Subsistema de Windows para Android™️ en Microsoft QA](https://learn.microsoft.com/es-es/answers/topics/windows-subsystem-for-android.html).

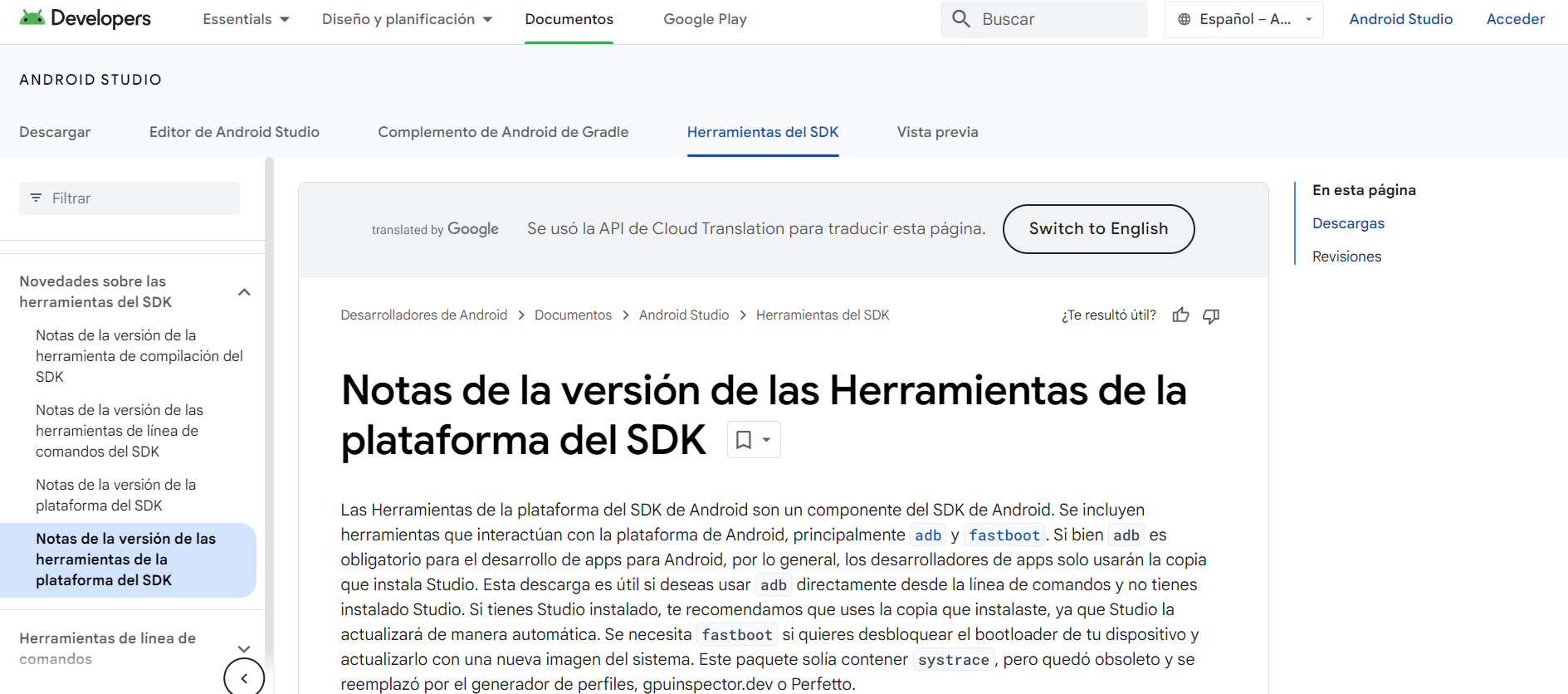
## Recursos adicionales

* [Aplicaciones de Amazon Appstore](https://support.microsoft.com/en-us/windows/apps-from-the-amazon-appstore-abed2335-81bf-490a-92e5-fe01b66e5c48)
* [Accesibilidad en el Subsistema de Windows para Android™️](https://support.microsoft.com/windows/97b77924-6df9-4597-8394-15b6e34d1658)

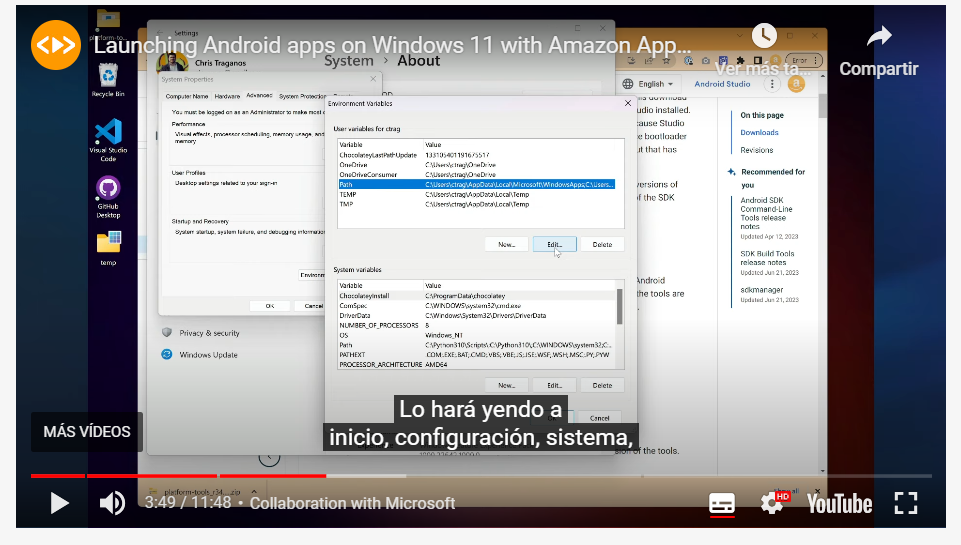




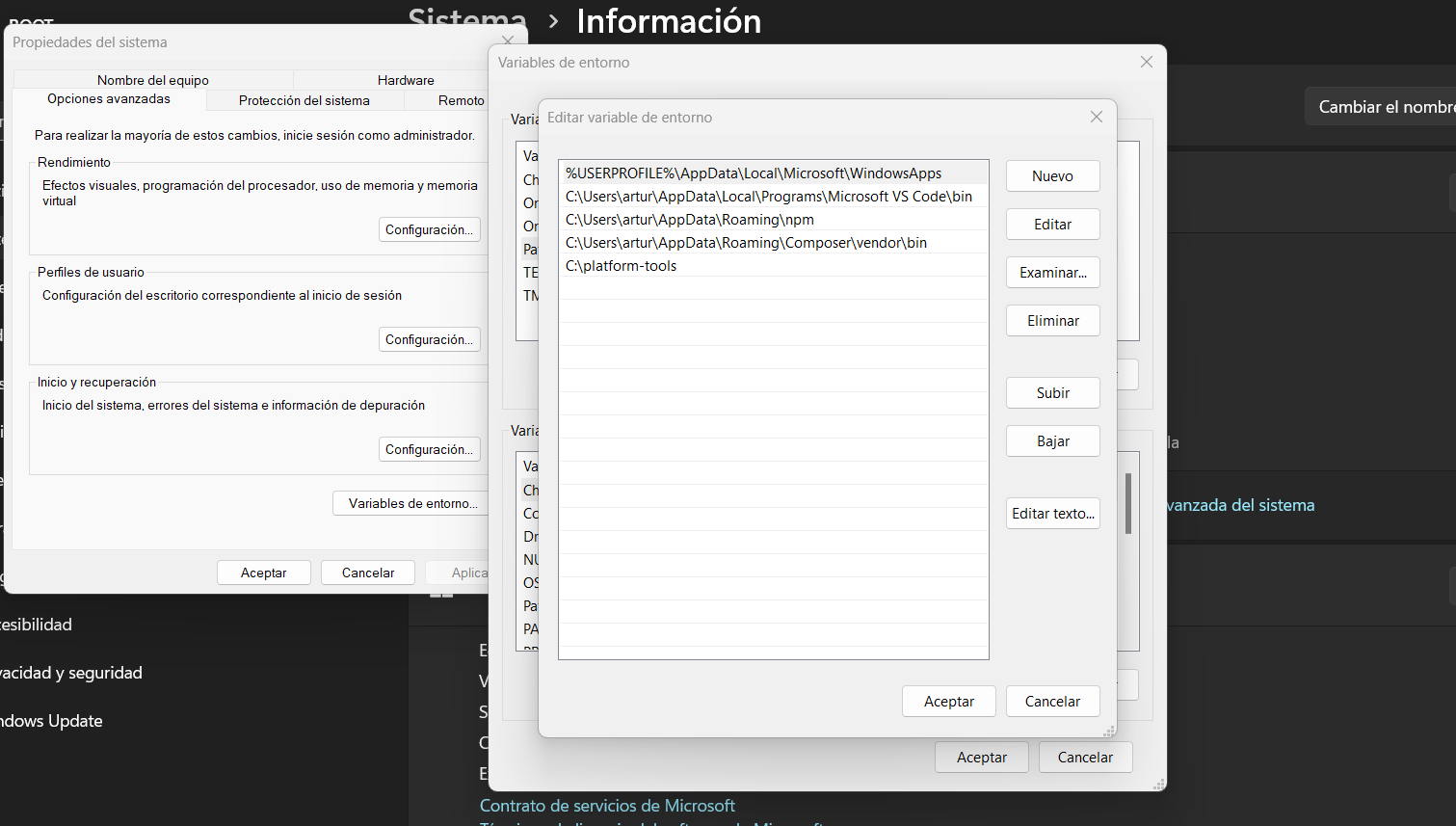


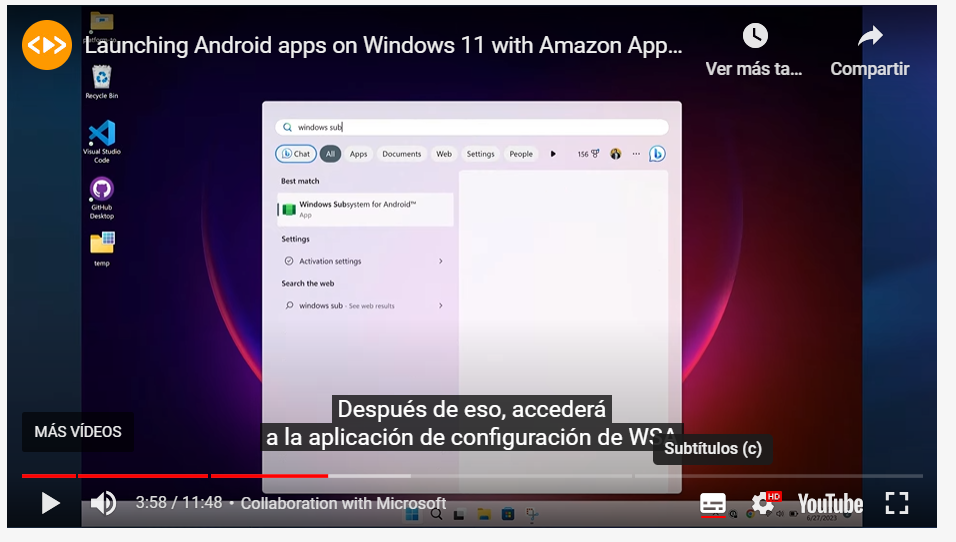




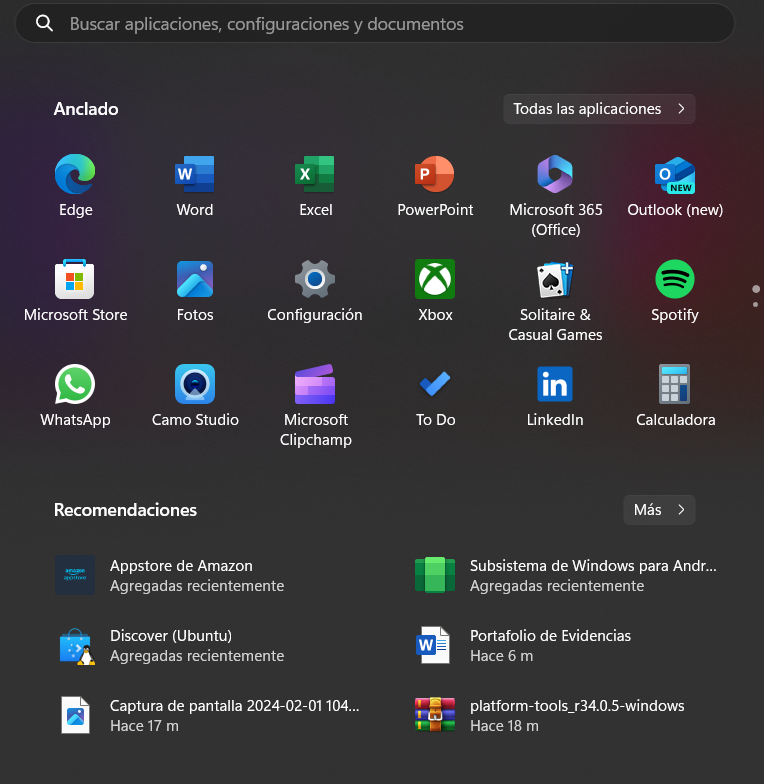


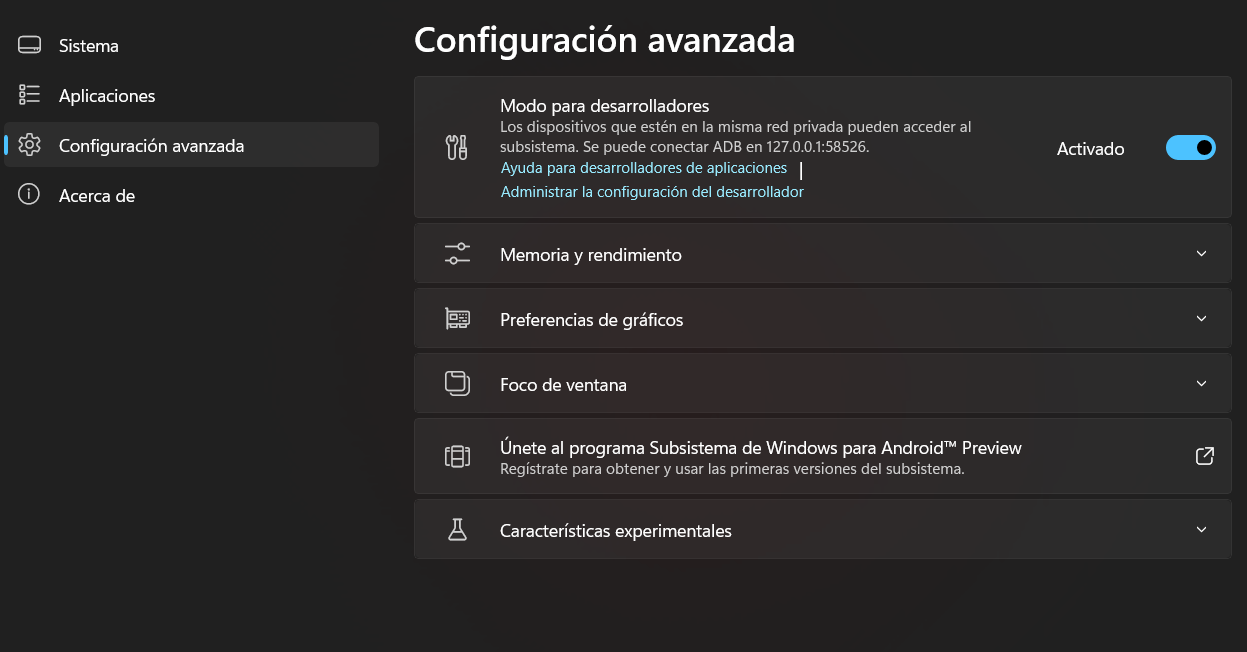


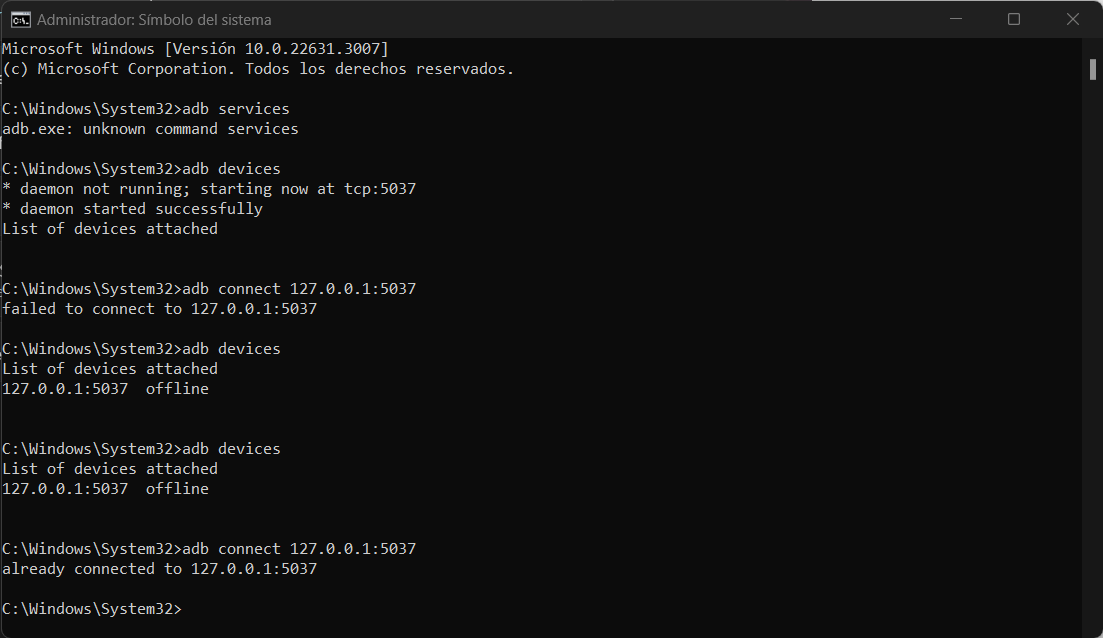


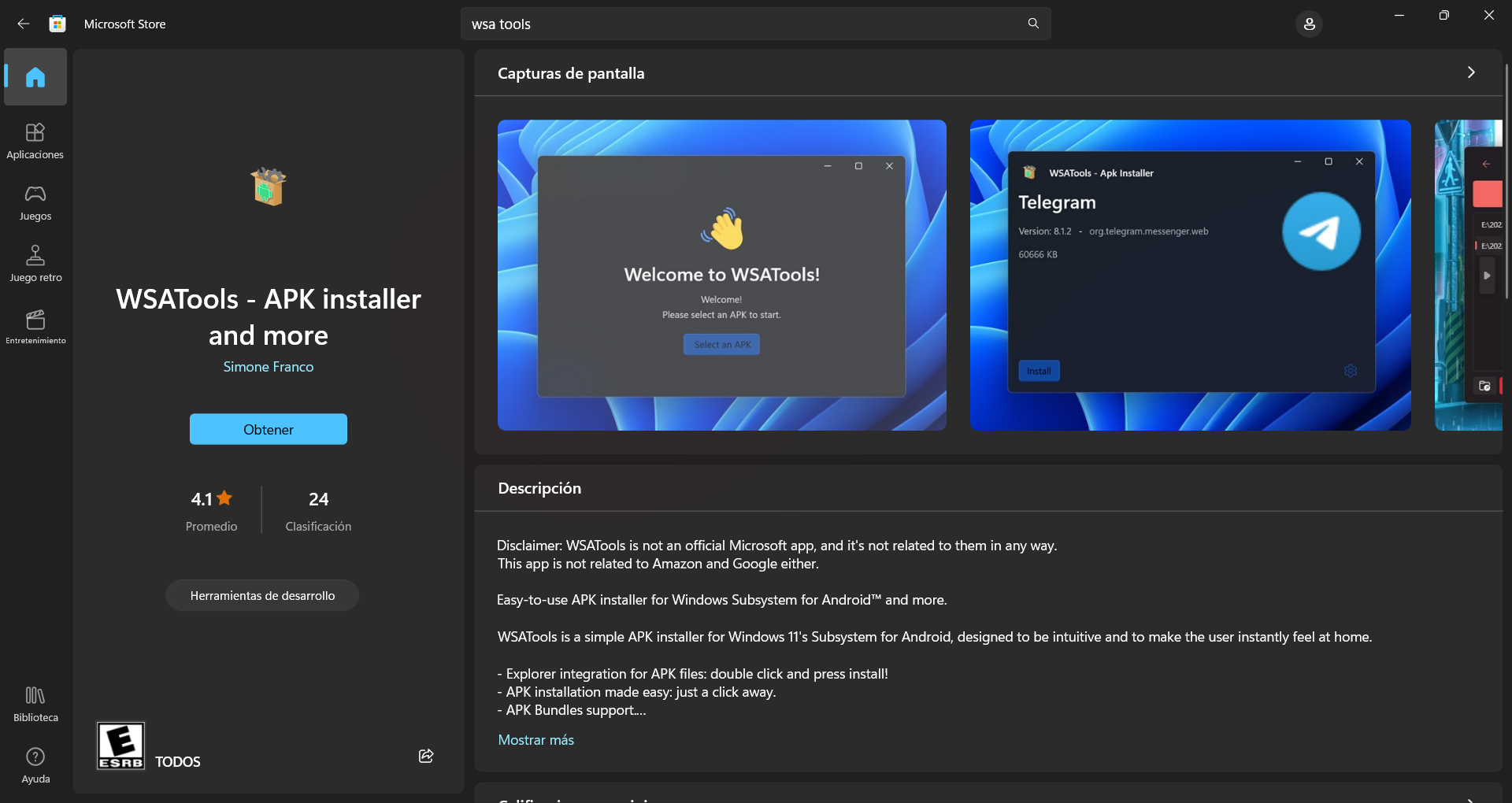


Para esto hay que reiniciar el equipo









# Clasificación de Sistemas Operativos

Los sistemas operativos modernos se pueden clasificar en varias categorías según su estructura y funcionalidades. Aquí hay una clasificación genérica junto con las estructuras comunes que se encuentran en los sistemas operativos actuales:

Clasificación:

Sistemas Operativos de Escritorio:

Diseñados para computadoras personales y estaciones de trabajo.

Ejemplos: Windows, macOS, Linux (varias distribuciones como Ubuntu, Fedora, etc.).

Sistemas Operativos Móviles:

Orientados a dispositivos móviles como teléfonos inteligentes y tabletas.

Ejemplos: Android, iOS.

Sistemas Operativos de Servidores:

Optimizados para entornos de servidor, ofrecen capacidad para manejar múltiples solicitudes de red y procesos en paralelo.

Ejemplos: Windows Server, Linux (varias distribuciones como CentOS, Debian, etc.).

Sistemas Operativos Embebidos:

Integrados en dispositivos y sistemas embebidos, como electrodomésticos, automóviles, dispositivos médicos, etc.

Ejemplos: FreeRTOS, Embedded Linux, Windows Embedded Compact.

Estructuras Genéricas:

Monolítico:

La mayoría de los componentes del sistema operativo residen en el espacio de núcleo.

Ejemplos: versiones más antiguas de Unix, MS-DOS.

Basado en Microkernel:

El núcleo proporciona solo las funciones más básicas, mientras que la mayoría de los servicios del sistema operativo se ejecutan como procesos fuera del núcleo.

Ejemplos: QNX, MINIX, GNU Hurd.

Híbrido:

Combina aspectos de los sistemas monolíticos y basados en microkernel.

Algunos componentes esenciales se ejecutan en el espacio de núcleo, mientras que otros se ejecutan como procesos separados.

Ejemplos: Linux, Windows NT/2000/XP/7/8/10.

Máquina Virtual:

Proporciona una capa de abstracción entre el hardware y el software del sistema operativo.

Ejemplos: Java Virtual Machine (JVM) que ejecuta sistemas operativos Java como JNode.

Observaciones adicionales:

Los sistemas operativos modernos a menudo incorporan características de seguridad avanzadas como la virtualización, controles de acceso, y cifrado de datos.

Muchos sistemas operativos de servidor y embebidos son distribuciones de Linux, debido a su naturaleza de código abierto y flexibilidad.

Los sistemas operativos móviles se están volviendo cada vez más importantes en el panorama tecnológico, con Android y iOS dominando el mercado.

Esta clasificación y estructuras genéricas proporcionan una visión general de la diversidad y complejidad de los sistemas operativos vigentes.

Los sistemas operativos se pueden clasificar de acuerdo a diferentes criterios:

Por número de usuarios:

Monousuario: Solo permiten un usuario a la vez. Un ejemplo clásico es MS-DOS.

Multiusuario: Permiten que varios usuarios trabajen al mismo tiempo. Algunos ejemplos son Windows, Linux y macOS.

Por número de tareas:

Monotarea: Solo permiten ejecutar una tarea a la vez.

Multitarea: Permiten ejecutar varias tareas al mismo tiempo. La mayoría de los sistemas operativos modernos son multitarea.

Por número de procesadores:

Uniprocesador: Diseñados para funcionar en un solo procesador.

Multiprocesador: Diseñados para funcionar en equipos con múltiples procesadores.

Por tipo de dispositivo:

Escritorio: Diseñados para ordenadores personales. Algunos ejemplos son Windows, macOS y Linux.

Móviles: Diseñados para teléfonos inteligentes y tablets. Algunos ejemplos son Android e iOS.

Embebidos: Diseñados para dispositivos específicos, como relojes inteligentes, televisores o automóviles.

Por estructura interna:

Monolítico: Todos los componentes del sistema operativo están en un solo bloque. Son más simples de implementar, pero menos flexibles y escalables.

Jerárquico: El sistema operativo se divide en diferentes capas, cada una con una función específica. Son más complejos de implementar, pero más flexibles y escalables.

Cliente-servidor: El sistema operativo se divide en dos partes: un núcleo (kernel) que proporciona los servicios básicos y servidores que proporcionan servicios específicos. Son más modulares y fáciles de mantener.

Estructuras genéricas de los Sistemas Operativos vigentes

Las estructuras genéricas más comunes de los sistemas operativos vigentes son:

Monolitica: Es la estructura más simple, donde todos los componentes del sistema operativo están en un solo bloque. Esta estructura es fácil de implementar, pero es menos flexible y escalable.

Jerárquica: El sistema operativo se divide en diferentes capas, cada una con una función específica. Esta estructura es más compleja de implementar, pero es más flexible y escalable. Algunos ejemplos de sistemas operativos con estructura jerárquica son Windows, Linux y macOS.

Cliente-servidor: El sistema operativo se divide en dos partes: un núcleo (kernel) que proporciona los servicios básicos y servidores que proporcionan servicios específicos. Esta estructura es modular y fácil de mantener. Algunos ejemplos de sistemas operativos con estructura cliente-servidor son Android e iOS.

Ejemplos de Sistemas Operativos

Algunos ejemplos de sistemas operativos vigentes son:

Escritorio: Windows, macOS, Linux, ChromeOS.

Móviles: Android, iOS, Windows Phone.

Embebidos: Android Things, FreeRTOS, QNX.