

## Manual técnico ABC Jobs

### Mobile

## Manual de Usuario de ABC Jobs

### Índice

1. Introducción.
2. JDK.
3. Instalación JDK.
4. SDK.
5. instalación SDK.
6. ADB.
7. Cómo instalar ADB.
8. Cómo compilar la aplicación ABC Job 's.
9. Cómo generar archivos APK para instalación manual.

## 1. Introducción

En este manual técnico, se encuentran detallados los procedimientos necesarios para llevar a cabo el despliegue e implementación de nuestro proyecto móvil. Estos elementos son fundamentales para garantizar el correcto funcionamiento de nuestra iniciativa, ya que de su adecuada ejecución y aplicación depende el éxito de nuestro proyecto.

## 2. JDK

El JDK, o "Java Development Kit" (Kit de Desarrollo de Java), es un conjunto de herramientas y software que se utiliza para desarrollar aplicaciones en el lenguaje de programación Java. El JDK incluye varios componentes esenciales para el desarrollo de software Java, que incluyen:

- **Compilador Java (javac):** El compilador Java traduce el código fuente escrito en Java a código de bytes Java (archivos .class) que puede ser ejecutado en la plataforma Java.
- **JRE (Java Runtime Environment):** El JDK incluye una versión del JRE, que es necesario para ejecutar aplicaciones Java en una máquina local. Esto permite probar y depurar las aplicaciones Java en el entorno de desarrollo antes de distribuirlas.
- **Herramientas de depuración:** El JDK proporciona herramientas para depurar código Java, como el depurador de línea de comandos (jdb) y herramientas de perfilado para analizar el rendimiento de las aplicaciones Java.
- **Bibliotecas y APIs:** El JDK incluye bibliotecas y APIs (Interfaces de Programación de Aplicaciones) que brindan acceso a funciones y clases estándar de Java, lo que facilita el desarrollo de aplicaciones.
- **Herramientas adicionales:** El JDK también incluye una variedad de herramientas adicionales, como el generador de documentación (javadoc) para crear documentación de código, herramientas de empaquetado para crear archivos JAR y mucho más.

## 3. Instalación JDK

El proceso para instalar el JDK (Java Development Kit) puede variar ligeramente según el sistema operativo que estés utilizando, ya sea Windows, macOS o Linux. A continuación, te proporciono los pasos generales para instalar el JDK en cada uno de estos sistemas:

## Instalación en Windows:

### Descargar el JDK:

- Visita el sitio web oficial de Oracle o del proveedor de JDK que desees utilizar (por ejemplo, OpenJDK) para descargar la versión más reciente del JDK compatible con Windows.

### Ejecutar el instalador:

- Ejecuta el archivo de instalación descargado.
- Sigue las instrucciones del instalador, que te guiarán a través del proceso de instalación.

### Configuración de variables de entorno (opcional):

- Después de la instalación, puedes configurar las variables de entorno para que el sistema reconozca la ubicación del JDK. Agrega la ruta al directorio "bin" del JDK a la variable de entorno "PATH".

### Verificación de la instalación:

- Abre una ventana de línea de comandos y ejecuta el comando `java -version` para verificar que el JDK se haya instalado correctamente. Deberías ver la información sobre la versión de Java instalada.

## Instalación en macOS:

### Descargar el JDK:

- Descarga el JDK desde el sitio web oficial de Oracle o del proveedor de JDK de tu elección. Asegúrate de descargar la versión compatible con macOS.

### Ejecutar el instalador:

- Ejecuta el archivo de instalación descargado.
- Sigue las instrucciones del instalador para completar el proceso de instalación.

### Configuración de variables de entorno (opcional):

- En macOS, generalmente no es necesario configurar las variables de entorno manualmente, ya que el sistema las configura automáticamente.

### Verificación de la instalación:

- Abre una terminal y ejecuta `java -version` para verificar que el JDK se haya instalado correctamente.

Instalación en Linux (Ubuntu como ejemplo):

**Usar el gestor de paquetes (recomendado):**

- En la mayoría de las distribuciones de Linux, puedes instalar el JDK utilizando el gestor de paquetes. En Ubuntu, puedes usar el siguiente comando:

```
sudo apt-get install default-jdk
```

**Configuración de variables de entorno (opcional):**

- Al igual que en macOS, en muchas distribuciones de Linux, no es necesario configurar las variables de entorno manualmente.

**Verificación de la instalación:**

- Abre una terminal y ejecuta `java -version` para verificar que el JDK se haya instalado correctamente.

## 4. SDK

El término "SDK" se refiere a un "Software Development Kit" o "Kit de Desarrollo de Software" en español. Un SDK es un conjunto de herramientas, bibliotecas y recursos que los desarrolladores utilizan para crear aplicaciones de software para una plataforma específica, sistema operativo o entorno de desarrollo.

Los SDKs proporcionan a los desarrolladores las herramientas necesarias para desarrollar, probar y depurar aplicaciones de manera más eficiente y efectiva. Estos kits pueden incluir:

- Herramientas de desarrollo: Esto puede incluir un compilador, un depurador y otras utilidades que permiten escribir y depurar código de manera efectiva.
- Bibliotecas y APIs: Los SDKs a menudo incluyen bibliotecas de funciones y APIs que simplifican tareas comunes y brindan acceso a características específicas de la plataforma o del sistema operativo.
- Documentación: Documentación detallada que explica cómo utilizar las herramientas y las bibliotecas proporcionadas en el SDK.
- Ejemplos de código: Ejemplos de código fuente que muestran cómo utilizar las bibliotecas y las funciones del SDK en aplicaciones reales.
- Emuladores o simuladores: Estos son programas que permiten probar aplicaciones en un entorno simulado antes de ejecutarlas en hardware real.
- Recursos adicionales: A veces, los SDKs pueden incluir recursos como imágenes, sonidos o plantillas que son útiles para el desarrollo de aplicaciones.

Los SDKs son particularmente importantes en el desarrollo de software para plataformas específicas, como aplicaciones móviles (Android SDK o iOS SDK), videojuegos (Unity SDK), desarrollo web (JavaScript SDKs), desarrollo de hardware (Arduino SDK), y muchas otras áreas. Proporcionan a los desarrolladores un punto de partida sólido y las herramientas necesarias para crear aplicaciones que aprovechen las características y funcionalidades de la plataforma o sistema para el que están destinadas.

## 5. Instalación de SDK

La instalación de un SDK (Software Development Kit) puede variar significativamente dependiendo de la plataforma o tecnología específica para la que esté diseñado el SDK. A continuación, te proporciono una descripción general de los pasos generales que se suelen seguir para instalar un SDK en un sistema:

### **Descarga el SDK:**

Visita el sitio web oficial del proveedor del SDK o la fuente confiable donde está disponible el SDK que deseas instalar. Esto podría ser el sitio web del desarrollador, un repositorio de software o una tienda de aplicaciones.

**Selecciona la versión adecuada:**

Asegúrate de descargar la versión del SDK que sea compatible con tu sistema operativo y la plataforma de desarrollo que estás utilizando. Por ejemplo, si estás desarrollando aplicaciones móviles para Android, deberás descargar el Android SDK.

**Sigue las instrucciones de instalación:**

Una vez que hayas descargado el SDK, sigue las instrucciones proporcionadas en la documentación oficial o en el sitio web para instalarlo en tu sistema operativo. Esto puede implicar ejecutar un archivo de instalación, descomprimir un archivo ZIP o realizar otros pasos específicos según el SDK.

**Configura las variables de entorno (si es necesario):**

Algunos SDKs pueden requerir que configures variables de entorno para que tu sistema pueda encontrar y utilizar las herramientas y bibliotecas proporcionadas por el SDK. Consulta la documentación del SDK para obtener información sobre la configuración específica.

**Verifica la instalación:**

Para asegurarte de que el SDK se haya instalado correctamente, abre una terminal o línea de comandos y ejecuta comandos o utilidades proporcionadas por el SDK, si las hay. Esto puede incluir comandos como `sdk-version`, `sdk info`, o cualquier otro comando específico del SDK que te permita verificar la instalación.

**Configura tu entorno de desarrollo (si es necesario):**

Dependiendo del SDK y del entorno de desarrollo que estés utilizando (por ejemplo, un IDE como Eclipse, Android Studio, Visual Studio, etc.), es posible que debas configurar tu entorno para que reconozca y utilice el SDK. Esto generalmente se hace a través de la configuración de las rutas o ubicaciones del SDK en la configuración del entorno de desarrollo.

Es importante mencionar que la instalación de un SDK puede variar según la plataforma y la tecnología específica. Siempre consulta la documentación oficial del SDK y sigue las instrucciones proporcionadas por el proveedor para asegurarte de realizar una instalación correcta y funcional.

## 6. ADB

ADB es una abreviatura que significa "Android Debug Bridge" (Puente de Depuración de Android, en español). Es una herramienta de línea de comandos que forma parte del Android SDK (Software Development Kit) y que se utiliza para interactuar con dispositivos Android desde un ordenador. ADB es una herramienta muy versátil y esencial para el desarrollo y depuración de aplicaciones Android. Algunas de sus funciones más comunes incluyen:

- Instalación y desinstalación de aplicaciones: ADB permite instalar y desinstalar aplicaciones en dispositivos Android conectados al ordenador. Esto es útil para probar aplicaciones durante el desarrollo.
- Transferencia de archivos: ADB facilita la transferencia de archivos entre el ordenador y el dispositivo Android, lo que es útil para cargar archivos de prueba o datos en el dispositivo.
- Depuración de aplicaciones: ADB es fundamental para la depuración de aplicaciones Android. Permite iniciar y detener sesiones de depuración, establecer puntos de interrupción, ver registros y obtener información sobre el estado de la aplicación en ejecución.
- Acceso a la consola del sistema: ADB permite acceder a la consola del sistema del dispositivo Android, lo que permite ejecutar comandos en el dispositivo desde el ordenador.
- Reiniciar y apagar el dispositivo: Puedes reiniciar o apagar el dispositivo Android conectado utilizando comandos ADB.
- Captura de pantalla: ADB puede tomar capturas de pantalla de la pantalla del dispositivo Android y guardarlas en el ordenador.
- Grabación de pantalla: ADB también puede ser utilizado para grabar la pantalla del dispositivo Android y guardar el vídeo resultante en el ordenador.

Para utilizar ADB, debes tener el Android SDK instalado en tu ordenador y habilitar las opciones de desarrollador en tu dispositivo Android. Una vez que hayas hecho esto y conectado tu dispositivo al ordenador, puedes abrir una ventana de línea de comandos y ejecutar comandos ADB para interactuar con el dispositivo.



Es importante mencionar que ADB es una herramienta potente que debe utilizarse con precaución, ya que puede tener acceso completo al dispositivo Android y sus datos. Por lo tanto, se utiliza principalmente con fines de desarrollo y depuración, y se recomienda no utilizarlo de manera indiscriminada en dispositivos no confiables o desconocidos.

## 7. Como instalar ADB

Para instalar ADB (Android Debug Bridge), debes seguir estos pasos generales. La forma exacta de instalarlo puede variar según tu sistema operativo, por lo que a continuación te proporciono instrucciones para las tres plataformas más comunes: Windows, macOS y Linux.

### En Windows:

#### 1. Descargar el paquete SDK Platform-Tools:

Visita el sitio web de descargas de Android (<https://developer.android.com/studio#downloads>) y busca la sección "Command line tools only". Descarga el paquete SDK Platform-Tools para Windows.

#### 2. Descomprimir el archivo descargado:

Descomprime el archivo ZIP que has descargado en una ubicación de tu elección. Por ejemplo, puedes descomprimirlo en la carpeta "C:\adb".

#### 3. Agregar la carpeta "platform-tools" a la variable de entorno PATH:

- Abre el menú Inicio y busca "Editar variables de entorno del sistema".
- Haz clic en "Editar las variables de entorno del sistema".
- En la sección "Variables de sistema", selecciona la variable "Path" y haz clic en "Editar...".
- Haz clic en "Nuevo" y agrega la ruta completa de la carpeta "platform-tools" que descomprimos en el paso 2 (por ejemplo, "C:\adb\platform-tools").
- Acepta los cambios y cierra las ventanas de configuración.

#### 4. Verificar la instalación:

Abre una ventana de línea de comandos y ejecuta el comando `adb version`. Deberías ver la información sobre la versión de ADB que se ha instalado.

#### En macOS:

##### 1. Instalar Homebrew (si aún no está instalado):

- Abre una terminal y ejecuta el siguiente comando para instalar Homebrew, un gestor de paquetes para macOS:

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install)"
```

##### 2. Instalar ADB:

- En la terminal, ejecuta el siguiente comando para instalar ADB a través de Homebrew:

```
brew install android-platform-tools
```

##### 3. Verificar la instalación:

En la terminal, ejecuta el comando `adb version` para verificar que ADB se haya instalado correctamente.

#### En Linux (Ubuntu como ejemplo):

##### 1. Instalar ADB desde el repositorio oficial:

Abre una terminal y ejecuta los siguientes comandos para instalar ADB desde el repositorio oficial de Ubuntu:

```
sudo apt-get update  
sudo apt-get install android-tools-adb
```

## **2. Verificar la instalación:**

En la terminal, ejecuta el comando `adb version` para verificar que ADB se haya instalado correctamente.

Después de seguir estos pasos, ADB debería estar instalado y listo para su uso en tu sistema. Puedes utilizar comandos ADB para interactuar con dispositivos Android conectados a tu ordenador.

## **8. Cómo compilar aplicación ABC Job's**

Para instalar una aplicación Android en un dispositivo o emulador utilizando Android Studio, sigue estos pasos:

### **1. Abre el proyecto en Android Studio:**

Abre Android Studio y carga el proyecto ABC Job's

### **2. Conecta el dispositivo o inicia el emulador:**

Asegúrate de que tu dispositivo Android esté conectado a tu ordenador a través de un cable USB y tenga la depuración USB habilitada. Si estás utilizando un emulador, inícialo desde Android Studio.

### **3. Selecciona el dispositivo de destino:**

En Android Studio, en la parte superior de la ventana, verás una barra de herramientas. Junto a la barra de herramientas, deberías ver una lista desplegable con una lista de dispositivos conectados o emuladores disponibles. Selecciona el dispositivo o emulador en el que deseas instalar la aplicación.

### **4. Compila y construye el proyecto:**

Asegúrate de que tu proyecto esté configurado correctamente y que no haya errores de compilación. Haz clic en "Build" en la barra de herramientas y selecciona "Build Project" para compilar tu aplicación.

## **5. Ejecuta la aplicación:**

Una vez que la compilación haya finalizado sin errores, haz clic en "Run" en la barra de herramientas y selecciona "Run 'app'". Esto iniciará la instalación de la aplicación en el dispositivo o emulador seleccionado.

## **6. Espera a que la aplicación se instale y se ejecute:**

Android Studio compilará la aplicación y la instalará en el dispositivo o emulador. Luego, la aplicación se ejecutará automáticamente en el dispositivo o emulador.

## **7. Verifica la ejecución de la aplicación:**

Asegúrate de que la aplicación se esté ejecutando correctamente en el dispositivo o emulador seleccionado. Puedes interactuar con la aplicación como lo harías con cualquier otra aplicación Android.

## **8. Depura y ajusta según sea necesario:**

Si encuentras problemas o errores en tu aplicación, puedes utilizar las herramientas de depuración de Android Studio para identificar y solucionar problemas. Esto incluye la capacidad de establecer puntos de interrupción, inspeccionar variables y más.

Eso es básicamente todo lo que necesitas hacer para instalar y ejecutar una aplicación Android en Android Studio. Recuerda que debes tener configurado el entorno de desarrollo correctamente, incluyendo la configuración de tu dispositivo Android o el uso de un emulador, antes de seguir estos pasos. También es importante que tu aplicación esté libre de errores de compilación y lógicos antes de intentar ejecutarla.

## **9. Cómo generar archivos APK para instalación manual.**

Para generar un archivo APK (Android Package) que puedas instalar manualmente en dispositivos Android, sigue estos pasos utilizando Android Studio:

### **1. Abre tu proyecto en Android Studio:**

Abre Android Studio y carga el proyecto de ABC Job's

### **2. Asegúrate de que tu proyecto esté listo:**

Antes de generar el APK, asegúrate de que tu proyecto esté configurado correctamente y que no haya errores de compilación.

### **3. Compila y construye el proyecto:**

Haz clic en "Build" en la barra de herramientas de Android Studio y selecciona "Build Bundle(s) / APK(s)" y luego "Build APK(s)".

### **4. Espera a que se complete la construcción:**

Android Studio generará uno o más archivos APK que representan tu aplicación. La ubicación predeterminada donde se almacenan los APK generados es en la carpeta `app/build/outputs/apk/` dentro de tu proyecto.

### **5. Encuentra el archivo APK generado:**

Ve a la carpeta `app/build/outputs/apk/` de tu proyecto en el explorador de archivos de tu ordenador. Allí encontrarás los archivos APK generados. Por lo general, habrá un APK "debug" y un APK "release".

### **6. Copia el APK a tu dispositivo Android:**

Transfiere el archivo APK que deseas instalar manualmente a tu dispositivo Android. Puedes hacerlo a través de USB, correo electrónico, almacenamiento en la nube u otras opciones de transferencia de archivos.

### **7. Habilita la instalación de aplicaciones desde fuentes desconocidas:**

Antes de instalar el APK manualmente, debes habilitar la opción para instalar aplicaciones desde fuentes desconocidas en tu dispositivo Android. Ve a la configuración de seguridad de tu dispositivo y activa la opción "Fuentes desconocidas" o "Orígenes desconocidos".

## **8. Instala el APK:**

Abre el administrador de archivos en tu dispositivo Android y busca el archivo APK que transferiste. Toca el archivo APK y sigue las instrucciones para instalar la aplicación en tu dispositivo.

## **9. Desactiva la opción "Fuentes desconocidas" (opcional):**

Por razones de seguridad, es recomendable desactivar la opción "Fuentes desconocidas" después de instalar la aplicación si no planeas instalar más aplicaciones manualmente desde fuentes no verificadas.

Una vez completados estos pasos, la aplicación debería estar instalada en tu dispositivo Android y lista para su uso. Ten en cuenta que este método se utiliza principalmente para pruebas o distribución de aplicaciones fuera de las tiendas de aplicaciones oficiales y debe realizarse con precaución para evitar la instalación de aplicaciones maliciosas.