**¿Por qué las personas eligen Java?**

Hay muchas razones por las cuales las personas eligen Java. Veamos algunos de los más comunes:

* Simplicidad: Java ofrece una experiencia bastante fácil para desarrolladores y usuarios finales. Esto se ha considerado consistentemente como una de las mayores ventajas de Java, especialmente cuando se compara con otros lenguajes de diseño de software. Java ha eliminado la importancia de los indicadores y ha cambiado la dificultad de las múltiples tradiciones en C ++, una estructura sencilla conocida popularmente como la interfaz.
* Portabilidad: Java es capaz de ejecutarse en cualquier momento / en cualquier lugar. La aplicación avanzada con la ayuda del lenguaje podría ejecutarse en cualquier plataforma de software y hardware.
* Asignación: Java tiene una característica del sistema de provisión de la pila que ayuda a las estadísticas a restablecerse rápidamente. A diferencia de otros lenguajes de desarrollo web, Java tiene el potencial de recolección automática de basura y distribución de memoria.
* Distributividad: la plataforma tiene una gran capacidad de networking. Esto significa que interactuar en Java es informal; escritura de programas de redes de sentidos como recibir y enviar archivos.
* Extremadamente seguro: Java es reconocido por sus estándares de seguridad y programación segura. Las compañías de desarrollo de Java pueden descargar cualquier carpeta con programas no confiables y luego la aplicación puede usar estos códigos no seguros de una manera segura.

Con sus raíces profundamente arraigadas en las comunidades de código abierto, las empresas y fundaciones como Apache, Google y otros gigantes de la industria popular han logrado contribuir con sus vastas bibliotecas de código e información.

Durante la conferencia I/O de Google en 2014, Google introdujo el visor Cardboard VR, un dispositivo barato hecho de cartón, el cuál usa lentes y un teléfono del usuario para proporcionar acceso simple a aplicaciones de realidad virtual. Dos años después, Google anunció planes para expandir esta plataforma vendiendo un visor más duradero con un control, conocido como Daydream View, el cuál es construido siguiendo el mismo concepto de usar un teléfono como el principal proveedor de RV. Para tener más aplicaciones desarrolladas que soporten esta plataforma, Google liberó SDKs de Cardboard para Android (SDK estándar y el NDK), iOS, el motor de juegos Unity y el motor de juegos Unreal. La Realidad Virtual (VR) es una de las tecnologías emergentes más discutidas de los últimos años. Especialmente en 2016, que marcó el lanzamiento de los primeros auriculares de realidad virtual para consumidores. Los desarrolladores de juegos de todo el mundo están haciendo juegos geniales para la realidad virtual y las empresas están buscando nuevas soluciones geniales con realidad virtual.

**El SDK para Android**

El SDK de Android permite el desarrollo de aplicaciones que, con el uso de las gafas del gran buscador, son capaces de mostrar imágenes en 3D que reaccionan ante el movimiento de la cabeza. El imán de las Cardboard, colocado en el lateral, permite interactuar con el smartphone Android, modificando el comportamiento de la brújula del teléfono y facilitando que la aplicación de realidad virtual funcione correctamente sin tocar el dispositivo.

[Según la información que facilita el propio buscador](https://developers.google.com/cardboard/android/), el SDK simplifica algunas tareas importantes para el desarrollo de la realidad virtual:

**- Corrección** de la distorsión de las gafas de realidad virtual.

**- Seguimiento** del movimiento de la cabeza del usuario.

**- Calibración** en 3D.

**- Renderización** en paralelo.

- Configuración de la **geometría** estereoscópica.

**- Manejo** de eventos de entrada del usuario.

**El SDK de Unity**

El SDK de Cardboard para Unity permite la creación de experencias de realidad virtual tanto para dispositivos Android como iOS ejecutadas en las gafas de cartón del gran buscador. Algunas de sus funciones permiten:

**- Crear de aplicaciones totalmente nuevas con Unity** o bien **adaptar una aplicación 3D hecha en esta plataforma a RV**.

**- Seguimiento del movimiento de la cabeza del usuario**.

**- Renderización estereoscópica en paralelo**.

**- Configuración estereoscópica automática** para un modelo concreto de Cardboard.

**- Corrección de distorsión para las Cardboard**.

**- Detección de eventos y entradas para la Cardboard**.

**- Corrección de desvío de giro automático**.

**- Configuración de los parámetros de los auriculares**.

**- Ajusta el nivel estereoscópico de forma dinámica** para reducir el cansancio ocular del usuario al usar las gafas de RV.

**- Determina la orientación de la mirada del usuario**.

**- Permite la interacción con elementos de la interfaz de usuario de Canvas** utilizando, por ejemplo, la mirada.

**- Respeta los efectos de zoom de la cámara**, compatibles con el movimiento de la cabeza del usuario.

Y sobre el tema encontré este video en YouTube de cerca de cincuenta minutos sobre el tema **Virtual Reality in Java, is it possible? by Erik Pronk & Alexander Chatzizacharias** https://www.youtube.com/watch?v=31v5EE1Loyc