

Prueba de aplicaciones web y para dispositivos móviles (Puesta en producción segura)

Fundamentos de la programación

El proceso de escribir código requiere frecuentemente conocimientos en varias áreas distintas, además del dominio del lenguaje a utilizar, algoritmos especializados y lógica formal.

Lenguajes de programación interpretados y compilados

La principal diferencia entre un lenguaje compilado y uno interpretado es que el lenguaje compilado requiere un paso adicional antes de ser ejecutado, la compilación, que convierte el código que escribes a lenguaje de máquina.

Código fuente y entornos de desarrollo

El código fuente es el conjunto de archivos que contienen órdenes o instrucciones que forman el programa. Para hacer cambios en un programa es necesario acceder al código fuente.

Las instrucciones son ejecutadas por la CPU que funciona del siguiente modo: obtiene la primera instrucción de la memoria, la decodifica para determinar el código de operación y los datos, después la ejecuta y en algunos casos almacena el resultado. Este proceso se ejecuta continuamente hasta que se leen todas las instrucciones del programa.

Se llama código objeto al código que resulta de la compilación del código fuente. Puede ser en lenguaje máquina o bytecode, y puede distribuirse en varios archivos que corresponden a cada código fuente compilado.

Luego un enlazador (linker) se encarga de juntar todos los archivos de código objeto para obtener el programa ejecutable.

Código objeto: Conjunto de instrucciones y datos escritos en un lenguaje que entiende el ordenador directamente: binario o código máquina.

Proviene de la traducción de cierto código fuente, es un fragmento del programa final y es específico de la plataforma de ejecución.

Una máquina virtual es una máquina virtual de proceso nativo, es decir, ejecutable en una plataforma específica, capaz de interpretar y ejecutar instrucciones expresadas en un código binario especial, el cual es generado por el compilador del lenguaje que estemos utilizando (por ejemplo Java).

Ejecución de software

Una vez que tenemos el código fuente escrito, hay que compilarlo mediante una herramienta de compilación específica para el lenguaje en el que estamos programando, entonces se genera un código objeto que puede ser en lenguaje máquina o bytecode y son entendidos por el sistema operativo.

También hay un elemento importante para considerar y es un enlazador (linker) se encarga de juntar todos los archivos de código objeto para obtener el programa ejecutable.

Elementos principales de los

programas

Elementos de la programación:

- Sintaxis.
- Código objeto.
- Depuración.
- Errores de sintaxis.
- Errores lógicos.
- Errores de regresión.
- Elementos léxicos de los programas.
- Identificador.

Pruebas. Tipos

Es fundamental verificar que los programas desarrollados cumplen con los requisitos planteados por los usuarios.

Hay que detectar los errores producidos durante el proceso de diseño y desarrollo y corregirlos.

La realización de pruebas permite verificar y validar el software.

Pruebas funcionales: una prueba funcional es una prueba basada en la ejecución, revisión y retroalimentación de las funcionalidades previamente diseñadas para el software (requisitos funcionales).

Hay distintos tipos como por ejemplo:

- Pruebas unitarias
- Pruebas de componentes
- Pruebas de integración
- Pruebas de sistema
- Pruebas de humo
- Pruebas alpha
- Pruebas beta

- Pruebas de aceptación
- Pruebas de regresión

Pruebas no funcionales: una prueba no funcional es una prueba cuyo objetivo es la verificación de un requisito que especifica criterios que pueden usarse para juzgar la operación de un sistema (requisitos no funcionales) como por ejemplo la disponibilidad, accesibilidad, usabilidad, mantenibilidad, seguridad, rendimiento.

Podemos clasificar las pruebas no funcionales según el tipo de requisito no funcional que abarcan:

- Pruebas de compatibilidad
- Pruebas de seguridad
- Pruebas de Estrés
- Pruebas de usabilidad
- Pruebas de rendimiento
- Pruebas de internacionalización y localización
- Pruebas de escalabilidad
- Pruebas de mantenibilidad
- Pruebas de instalabilidad
- Pruebas de portabilidad

Más información

- <https://www.jesusninoc.com/03/25/ejecutar-google-chrome-en-android-mediante-adb-a-traves-de-powershell/>
- <https://www.jesusninoc.com/03/25/abrir-una-pagina-web-mediante-adb-a-traves-de-powershell/>
- <https://www.jesusninoc.com/03/12/obtener-un-listado-de-las-aplicaciones-instaladas-en-android-con-adb/>
- <https://www.jesusninoc.com/03/11/conocer-la-ip-asignada-al-dispositivo-android-con-adb/>

Seguridad en los lenguajes de programación y sus entornos de ejecución («sandboxes»)

El término Sandbox o Sandboxing, tal y como indica su nombre, se refiere a un «cajón de arena» o un entorno controlado.