

Definiciones, primera clase programación, orientada ha objetos.

Edgar Mauricio Navarro Rocha código: 625171

1) **Java**: Java es toda una tecnología orientada al desarrollo de software con el cual podemos realizar cualquier tipo de programa. Hoy en día, la tecnología Java ha cobrado mucha importancia en el ámbito de Internet gracias a su plataforma J2EE. Pero Java no se queda ahí, ya que en la industria para dispositivos móviles también hay una gran acogida para este lenguaje.

La tecnología Java está compuesta básicamente por 2 elementos: el lenguaje Java y su plataforma.

Con plataforma nos referimos a la máquina virtual de Java (Java Virtual Machine).Java también es un lenguaje de programación. En la actualidad es un lenguaje muy extendido y cada vez cobra más importancia tanto en el ámbito de Internet como en la informática en general .Está desarrollado por la compañía Sun Microsystems, desarrollado por James Gosling en 1995, con gran dedicación y siempre enfocado a cubrir las necesidades tecnológicas más punteras.

l lenguaje en sí mismo toma mucha de su sintaxis de C, Cobol y Visual Basic, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria. La memoria es gestionada mediante un recolector de basura.

2) **UML**: EL LENGUAJE UNIFICADO DE MODELADO (UML), es un lenguaje para hacer modelos y es independiente de los métodos de análisis y diseño.

En todas las disciplinas de la Ingeniería se hace evidente la importancia de los modelos ya que describen el aspecto y la conducta de "*algo*". Ese "*algo*" puede existir, estar en un estado de desarrollo o estar, todavía, en un estado de planeación. Es en este momento cuando los diseñadores del modelo deben investigar los requerimientos del producto terminado y dichos requerimientos pueden incluir áreas tales como funcionalidad, performance y confiabilidad. Además, a menudo, el modelo es dividido en un número de vistas, cada una de las cuales describe un aspecto específico del producto o sistema en construcción.

El modelado sirve no solamente para los grandes sistemas, aun en aplicaciones de pequeño tamaño se obtienen beneficios de modelado, sin embargo es un hecho que entre más grande y más complejo es el sistema, más importante es el papel de que juega el modelado por una simple razón: "El hombre hace modelos de sistemas complejos porque no puede entenderlos en su totalidad".

UML es una técnica para la especificación sistemas en todas sus fases. Nació en 1994 cubriendo los aspectos principales de todos los métodos de diseño antecesores y, precisamente, los padres de UML son Grady Booch, autor del método Booch; James Rumbaugh, autor del método OMT e Ivar Jacobson, autor de los métodos OOSE y Objectory.

La versión 1.0 de UML fue liberada en Enero de 1997 y ha sido utilizado con éxito en sistemas construidos para toda clase de industrias alrededor del mundo: hospitales, bancos, comunicaciones, aeronáutica, finanzas, etc.

3) **C++**: es un lenguaje de programación diseñado a mediados de los años 1980 por Bjarne Stroustrup. La intención de su creación fue el extender al exitoso lenguaje de programación C con mecanismos que permitan la manipulación de objetos. En ese sentido, desde el punto de vista de los lenguajes orientados a objetos, el C++ es un lenguaje híbrido. Además es un lenguaje que sirve para ordenar alas computadoras.

4) **ORM**: es un poderoso método para el diseño de modelos de bases de datos y consulta a nivel conceptual, donde la aplicación se describe en términos fácilmente comprensibles por los usuarios no técnicos. En la práctica, los datos ORM modelos suelen captar más las reglas de negocio, y son más fáciles de validar y desarrollar modelos de datos que en otros enfoques.

5) **GRASP**: Generales de Responsabilidad patrones de software de asignación (o principios), GRASP abreviada, consiste en directrices para la asignación de responsabilidad de las clases y objetos en diseño orientado a objetos .

Los diferentes patrones y los principios utilizados en GRASP son: Controlador, el Creador, indirecto, de expertos de la Información , de alta cohesión , bajo acoplamiento , polimorfismo , Variaciones Protegidas, y la pura invención. Todos estos patrones de respuesta a algunas de software problema, y en casi todos los casos estos problemas son comunes a casi todos los de desarrollo de software del proyecto. Estas técnicas no se han inventado para crear nuevas formas de trabajar, sino para documentar mejor y estandarizar la edad, probadas y comprobadas de programación en los principios de diseño orientado a objetos.

Larman afirma que "la herramienta de diseño esencial para el desarrollo de software es una mente bien educada en los principios de diseño. No es el UML o cualquier otra tecnología". Por lo tanto, GRASP es en realidad un conjunto de herramientas mentales, una ayuda de aprendizaje para ayudar en la diseño de software orientado a objetos.

6) Solid: es un acrónimo inventado por Robert C.Martin para establecer los cinco principios básicos de la programación orientada a objetos y diseño. Este acrónimo tiene bastante relación con los patrones de diseño, en especial, con la alta cohesión y el bajo acoplamiento.

El objetivo de tener un buen diseño de programación es abarcar la fase de mantenimiento de una manera más legible y sencilla así como conseguir crear nuevas funcionalidades sin tener que modificar en gran medida código antiguo. Los costes de mantenimiento pueden abarcar el 80% de un proyecto de software por lo que hay que valorar un buen diseño.

S-Responsabilidad simple (Single responsibility)

Este principio trata de destinar cada clase a una finalidad sencilla y concreta. En muchas ocasiones estamos tentados a poner un método reutilizable que no tienen nada que ver con la clase simplemente porque lo utiliza y nos pilla más a mano. En ese momento pensamos “Ya que estamos aquí, para que voy a crear una clase para realizar esto. Directamente lo pongo aquí”.

El problema surge cuando tenemos la necesidad de utilizar ese mismo método desde otra clase. Si no se refactoriza en ese momento y se crea una clase destinada para la finalidad del método, nos toparemos a largo plazo con que las clases realizan tareas que no deberían ser de su responsabilidad.

Con la anterior mentalidad nos encontraremos, por ejemplo, con un algoritmo de formateo de números en una clase destinada a leer de la base de datos porque fue el primer sitio donde se empezó a utilizar. Esto conlleva a tener métodos difíciles de detectar y encontrar de manera que el código hay que tenerlo memorizado en la cabeza.

O-Abierto/Cerrado (Open/Closed)

Principio atribuido a Bertrand Meyer que habla de crear clases extensibles sin necesidad de entrar al código fuente a modificarlo. Es decir, el diseño debe ser abierto para poderse extender pero cerrado para poderse modificar. Aunque dicho parece fácil, lo complicado es predecir por donde se debe extender y que no tengamos que modificarlo. Para conseguir este principio hay que tener muy claro como va a funcionar la aplicación, por donde se puede extender y como van a interactuar las clases.

El uso más común de extensión es mediante la herencia y la reimplementación de métodos. Existe otra alternativa que consiste en utilizar métodos que acepten una **interfase** de manera que podemos ejecutar cualquier clase que

implemente esa interfase. En todos los casos, el comportamiento de la clase cambia sin que hayamos tenido que tocar código interno.

Como ya he comentado llega un momento en que las necesidades pueden llegar a ser tan imprevisibles que nos topemos que con los métodos definidos en las interfases o en los métodos extensibles, no sean suficientes para cubrir las necesidades. En este caso no habrá más remedio que romper este principio y refactorizar.

L-Sustitucion Liskov (Liskov substitution)

Este principio fue creado por Barbara Liskov y habla de la importancia de crear todas las clases derivadas para que también puedan ser tratadas como la propia clase base. Cuando creamos clases derivadas debemos asegurarnos de no reimplementar métodos que hagan que los métodos de la clase base no funcionasen si se tratasen como un objeto de esa clase base.

I-Segregacion del interface (Interface segregation)

Este principio fue formulado por Robert C. Martín y trata de algo parecido al primer principio. Cuando se definen interfaces estos deben ser específicos a una finalidad concreta. Por ello, si tenemos que definir una serie de métodos abstractos que debe utilizar una clase a través de interfaces, es preferible tener muchos interfaces que definan pocos métodos que tener un interface con muchos métodos.

El objetivo de este principio es principalmente poder reaprovechar los interfaces en otras clases. Si tenemos un interfase que compara y clona en el mismo interfase, de manera más complicada se podrá utilizar en una clase que solo debe comparar o en otra que solo debe clonar.

D-Inversión de dependencias (Dependency inversion)

También fue definido por Robert C. Martín. El objetivo de este principio conseguir desacoplar las clases. En todo diseño siempre debe existir un acoplamiento pero hay que evitarlo en la medida de lo posible. Un sistema no acoplado no hace nada pero un sistema altamente acoplado es muy difícil de mantener.

El objetivo de este principio es el uso de abstracciones para conseguir que una clase interactúe con otras clases sin que las conozca directamente. Es decir, las clases de nivel superior no deben conocer las clases de nivel inferior. Dicho de otro modo, no debe conocer los detalles. Existen diferentes patrones como la inyección de dependencias o service locator que nos permiten invertir el control.

7) **GitHub**: es un servicio basado en Web de alojamiento de proyectos de desarrollo de software que utilizan el Git de control de revisiones del sistema.

GitHub ofrece dos planes de pago para los depósitos privados, y las cuentas gratuitas para los proyectos de código abierto. En mayo de 2011, GitHub fue el más popular de código de código abierto sitio de repositorio.

8) **Git**: es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente. Al principio, Git se pensó como un motor de bajo nivel sobre el cual otros pudieran escribir la interfaz de usuario o front end como Cogito o StGIT.

Sin embargo, Git se ha convertido desde entonces en un sistema de control de versiones con funcionalidad plena. Hay algunos proyectos de mucha relevancia que ya usan Git, en particular, el grupo de programación del núcleo Linux.

9) **VCS**: Sistema *de control de versiones* es un software que permite gestionar los cambios de los documentos, programas, imágenes y otra información que se almacenan en forma de archivos informáticos.

Los cambios suelen ser identificados por un número que se incrementa o código de letra también se conoce como número de revisión o modificación.

El uso más simple de control de versiones es que, el usuario puede fácilmente volver a la versión anterior de trabajo de sus archivos, en caso de que fallan en algo, con los últimos cambios.

Los cambios podrían ir desde la fijación de un error en un archivo de texto hasta una refactorización muy importante en un proyecto de software, que abarca cientos de archivos. Cada cambio por lo general tiene el nombre de la persona que lo introdujo, el tiempo del cambio y un mensaje de descripción opcional.

10) **Sky Drive**: (oficialmente **Microsoft SkyDrive**) es un servicio de alojamiento de archivos que permite a los usuarios cargar y sincronizar archivos a un almacenamiento en la nube y acceder a ellos desde un navegador Web o de su dispositivo local. Es parte de la Windows Live gama de servicios en línea y permite a los usuarios para mantener los archivos privados, compartir con los contactos, o hacer que los archivos públicos. Los archivos compartidos públicamente no requieren un Windows Live ID para acceder.

11) **Drop Box:** es una de la más simple, más elegante de sincronización de archivos herramienta que hemos utilizado. La premisa detrás de Dropbox es que le da acceso a sus archivos sin importar qué computadora o dispositivo que tengas a mano. El servicio almacena los archivos con el cifrado fuerte en servidores múltiples y le permite llegar a sus archivos de forma rápida, fácil y en su mayor parte con elegancia desde prácticamente cualquier dispositivo habilitado para Internet.

Dropbox es un producto basado en web y descargables. Funciona igualmente sin problemas en Windows, Mac, Linux, así como los dispositivos móviles, incluyendo iOS (véase nuestra revisión de Dropbox para el iPad , por ejemplo), Android y BlackBerry. Dropbox sincroniza sólo los archivos almacenados en una carpeta dedicada única, así que si usted prefiere para sincronizar las carpetas que ya tiene en su sistema sin moverlos a la carpeta de Dropbox, Dropbox no puede ser para usted (trate de SugarSync en su lugar).

Pero su buen funcionamiento y sin problemas lo convierte en uno de las opciones de los editores para compartir archivos y soluciones de copia de seguridad.

12) **JDK: (Java Development Kit)** es un entorno de desarrollo de programas para la escritura de **Java applets** y aplicaciones de s. Se trata de un entorno de ejecución que "se sienta en la parte superior" del **sistema operativo de** la capa, así como las herramientas y la programación que los desarrolladores necesitan para compilar, depurar y ejecutar applets y aplicaciones escritas en lenguaje Java.

13) **JRE (Java Runtime Environment).**

Java Runtime Environment contiene la JVM, las bibliotecas de clases, y otros archivos auxiliares. No contiene ninguna herramienta de desarrollo, como compilador, depurador, etc.

En realidad JVM se ejecuta el programa, y utiliza las bibliotecas de clases, y otros archivos auxiliares previstos en el JRE. Si desea ejecutar cualquier programa en Java, es necesario tener JRE instalado en el sistema.

La máquina virtual de Java proporciona una forma independiente de la plataforma de la ejecución de código, los programadores pueden concentrarse en el software de grabación, sin tener que preocuparse por cómo o dónde se ejecutará.

Si u sólo desea ejecutar applets (por ejemplo: Los juegos en línea de Yahoo o rompecabezas), *JRE* debe ser instalado en la máquina.

14) **NOTEPAD ++** : es un editor de texto y el editor de código fuente para los de Windows . Una de las ventajas de Notepad + + en el built-in editor de texto de Windows, el Bloc de notas , es la edición de fichas ., lo que permite trabajar con varios archivos abiertos.

Notepad + + se distribuye como software libre . El proyecto se encuentra alojado en SourceForge.net , desde donde ha sido descargado más de 27 millones de veces y ganó dos veces la Comunidad SourceForge Choice Award por Mejor herramienta de desarrollo. [4] . El proyecto se encuentra alojado en TuxFamily desde junio de 2010. Para visualizar y editar texto y del lenguaje de programación de código fuente los archivos, Notepad + + utiliza el componente de edición Scintilla.

15) **WEB SERVICES**: Es un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones.

Los Web Services permite la comunicación entre aplicaciones o componentes de aplicaciones de forma estándar a través de protocolos comunes (como http) y de manera independiente al lenguaje de programación, plataforma de implantación, formato de presentación o sistema operativo. Un Web Service es un contenedor que encapsula funciones específicas y hace que estas funciones puedan ser utilizadas en otros servidores.

16) **WEB APPS**: Una aplicación web (aplicaciones web) es un programa de aplicación que se almacena en un servidor remoto y se entregan a través de Internet a través de una interfaz de navegador.

Los servicios Web son aplicaciones de la Web, por definición, y muchos, aunque no todos, los sitios web contienen aplicaciones de la Web. De acuerdo con Web.

AppStorm editor de Jarel Remick, cualquier componente de sitio web que realiza una función para el usuario califica como una aplicación web.

Una aplicación Web es una aplicación que se accede a través de una red como la internet o una intranet . El término también puede significar una aplicación de software que se codifica en un lenguaje de navegador compatible (como JavaScript , combinado con un navegador-reenderezados, lenguaje de marcado como HTML) y dependen de un navegador web común para hacer que la aplicación ejecutable.

17) **RCS**: El sistema de control de revisiones es una implementación software de control de revisión que automatiza la recuperación de almacenamiento, las revisiones de registro, identificación y fusión de. RCS es útil para el texto que se revisa con frecuencia, por ejemplo, programas , documentación, gráficos de procedimiento, papeles y cartas. RCS es también capaz de manejar archivos binarios, aunque con una eficacia reducida. Revisiones se almacenan con la ayuda de la diferencia de utilidad.

18) **CVS**: es un sistema de control de versiones. Con ella, usted puede registrar la historia de los archivos de origen.

Por ejemplo, los errores a veces, la fluencia cuando el software se modifica, y que puede no detectar el error hasta mucho tiempo después de realizar la modificación. Con CVS, usted puede recuperar fácilmente versiones anteriores para ver exactamente lo que el cambio causado el error. Esto a veces puede ser de gran ayuda.

Usted podría, por supuesto, salvar a todas las versiones de cada archivo que has creado. Sin embargo, esto perdería una enorme cantidad de espacio en disco. CVS almacena todas las versiones de un archivo en un único archivo de una manera inteligente que almacena sólo las diferencias entre versiones.

CVS comenzó como un montón de scripts de shell escrito por Dick Grune, publicado en comp.sources.unix en el volumen 6 liberación de diciembre de 1986. Si bien no hay ningún código real de estos scripts de shell está presente en la versión actual de CVS gran parte de los algoritmos de resolución de conflictos de CVS provenir de ellos.

19) **Mercurial**: es un sistema de control de versiones multiplataforma, que está implementado utilizando el lenguaje Python, pero con una implementación binaria de diff escrita en C.

En sus orígenes fue desarrollado para Linux, pero luego se han sacado versiones tanto para Windows como para Mac. Es sobre todo un programa

para línea de comandos y todas las operaciones se invocan como opciones dadas a su programa motor, hg, que hace referencia al símbolo químico del mercurio. Este pensado para el desarrollo integralmente distribuido y para poder funcionar sin un servidor, incluye una interfaz web integrada.

20) **Bazaar**: es un sistema de control de versiones que le ayuda a realizar un seguimiento de la historia del proyecto en el tiempo y para colaborar fácilmente con otros. Tanto si eres un desarrollador individual, un equipo de co-localizados o una comunidad de desarrolladores repartidos por todo el mundo, las escalas de bazar y se adapta a sus necesidades. Parte del Proyecto GNU.

21) **SVN**: es un sistema OpenSource de control concurrente de versiones. Como tal, permite la modificación simultánea de archivos por múltiples usuarios, automatizando la combinación de los cambios. La base de datos de SVN guarda la historia de todas las versiones, permitiendo así revisar éstas o revertir modificaciones.

22) **Amazon Web Services : (AWS** abreviada) es una colección de computación a distancia de servicios (también llamados servicios web) que forman parte de un cloud computing plataforma, que ofrece a través de Internet Amazon.com.

23) **EC2** : es una parte central de Amazon.com 's cloud computing plataforma, Amazon Web Services (AWS).

EC2 permite a los usuarios para alquilar ordenadores virtuales en los que se ejecuten sus propias aplicaciones informáticas . EC2 permite escalable despliegue de aplicaciones, proporcionando un servicio Web a través del cual un usuario puede iniciar un Amazon Machine Image para crear una máquina virtual , el cual Amazon llama a una "instancia", que contiene el software que desee. Un usuario puede crear, lanzar y poner fin a las instancias del servidor, según sea necesario, el pago por hora para los servidores activos, de ahí el término "elástico". EC2 proporciona a los usuarios el control sobre la ubicación geográfica de los casos que permite la optimización de la latencia y los altos niveles de redundancia.

24) **XML** : es un estándar, simple, auto-descripción de forma de codificar el texto y los datos para que el contenido puede ser procesado con la intervención humana relativamente poco e intercambiar a través de distinto hardware, sistemas operativos y aplicaciones.

En resumen, XML ofrece una forma ampliamente adoptado estándar de representar texto y datos en un formato que se pueden procesar sin mucha inteligencia humana o de la máquina. La información formateada en XML se puede intercambiar a través de plataformas, lenguajes y aplicaciones, y se puede utilizar con una amplia gama de herramientas de desarrollo y utilidades.

XML es bastante similar al HTML en su formato actual (ambos están estrechamente relacionados con el lenguaje de definición de SGML marcado que ha sido un estándar ISO desde 1986), para que quienes estén familiarizados con HTML bastante facilidad recoger conocimientos básicos de XML. Pero hay dos diferencias fundamentales:

- La separación de forma y contenido - HTML se compone principalmente de las etiquetas que definen la apariencia del texto, en formato XML las etiquetas definen en general la estructura y el contenido de los datos, con apariencia real especificado por una aplicación específica o una hoja de estilo asociada.
- XML es extensible - las etiquetas pueden ser definidas por individuos u organizaciones para alguna aplicación específica, mientras que el conjunto de etiquetas HTML estándar se define por la World Wide Web Consortium (W3C).

25) **PARADIGMA:** Un paradigma es un conjunto de reglas que "rigen" una determinada disciplina. Están "reglas" se asumen normalmente como "verdades incuestionables", porque son "tan evidentes" que se tornan transparentes para los que están inmersos en ellas. Como el aire para las personas o el agua para el pez.

Lo que se vive hoy son permanentes "cambios paradigmáticos", en la educación, la economía, los negocios, las empresas, la política. Es decir un permanente cambio de las reglas. Los que se anticipan a los cambios son los innovadores, aquellos que empiezan cuestionando los paradigmas.

26) **OCJP: Java Programmer. Oracle Certified Professional (Antes SCJP)**

Esta certificación es para programadores experimentados en el uso del lenguaje de programación Java. La obtención de esta certificación garantiza que el programador comprende la sintaxis básica y la estructura del lenguaje Java.

Para lograr esta certificación el candidato debe completar de forma satisfactoria un examen. No es necesario haber obtenido ninguna otra certificación antes de obtener la certificación OCJP de programador Java.

27) **VM:** Una máquina virtual (VM) es una implementación de software de un entorno informático en el que un sistema operativo (OS puede ser) o el programa instalado y ejecutado. La máquina virtual normalmente emula un entorno informático físico, pero las solicitudes de CPU , memoria, disco duro , red y otros recursos de hardware son manejados por una capa de virtualización que se traduce en estas solicitudes al hardware físico subyacente.

Máquinas virtuales se crean dentro de una capa de virtualización, como un hypervisor o de una plataforma de virtualización que se ejecuta en la parte superior de un cliente del sistema operativo o del servidor. Este sistema operativo se conoce como el sistema operativo anfitrión . La capa de virtualización se puede utilizar para crear muchos entornos de VM individuales, aislados.

28) **DRY:** El principio No te repitas (en inglés Don't Repeat Yourself o DRY, también conocido como Una vez y sólo una) es una filosofía de definición de procesos que promueve la reducción de la duplicación especialmente en computación. Según este principio toda pieza de información nunca debería ser duplicada debido a que la duplicación incrementa la dificultad en los cambios y evolución posterior, puede perjudicar la claridad y crear un espacio para posibles inconsistencias. Por "pieza de información" podemos entender, en un sentido amplio, desde datos almacenados en una base de datos pasando por el código fuente de un programa de software hasta llegar a información textual o documentación.

Cuando el principio DRY se aplica de forma eficiente los cambios en cualquier parte del proceso requieren cambios en un único lugar. Por el contrario, si algunas partes del proceso están repetidas por varios sitios, los cambios pueden provocar fallos con mayor facilidad si todos los sitios en los que aparece no se encuentran sincronizados.

29) **MOBILE DEV:** Desarrollo de aplicaciones móviles es el proceso por el cual el software de aplicación se desarrolló para el bajo consumo de energía de dispositivos portátiles , tales como asistentes digitales personales , los asistentes digitales para empresas o teléfonos móviles . Estas aplicaciones se pre-instalado en los teléfonos durante la fabricación, puede ser descargado por los clientes de diversas plataformas móviles de distribución de software o aplicaciones web entregados a través de HTTP que usan del lado del servidor o el procesamiento del lado del cliente (por ejemplo, Java Script) para proporcionar una "aplicación-como "La experiencia dentro de un navegador Web.

30) **SISTEMA:** Un conjunto de detalles los métodos , los procedimientos y rutinas establecidas o formulado para llevar a cabo una determinada actividad , realizar un servicio o resolver un problema .

31) **LANG:** La biblioteca de programación S-Lang es una biblioteca de software de Unix , de Windows , VMS , OS / 2 y Mac OS X . Proporciona rutinas para la incorporación de un intérprete para el lenguaje de scripting S-Lang, y los componentes para facilitar la creación de texto aplicaciones basadas en esta última clase de funciones incluyen rutinas para la construcción y manipulación de mapas de teclas, una instalación interactiva de línea de edición, y dos bajas y de alto nivel en la pantalla terminal de funciones de gestión.

