**JAVA:** Java es un lenguaje de programación de alto nivel orientado a objetos, desarrollado por James Gosling en 1995. El 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.

**VML:**Vector Markup Language (VML) es un lenguaje XML de programación abierto, destinado a la creación de los gráficos vectoriales elaborados en 2D o 3D (estáticos o animados) en las páginas Web.

**C++:** 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.

**ORM:** El mapeo objeto-relacional (más conocido por su nombre en inglés, Object-Relational mapping, o sus siglas O/RM, ORM, y O/R mapping) es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional, utilizando un motor de persistencia. En la práctica esto crea una base de datos orientada a objetos virtual, sobre la base de datos relacional. Esto posibilita el uso de las características propias de la orientación a objetos (básicamente herencia y polimorfismo). Hay paquetes comerciales y de uso libre disponibles que desarrollan el mapeo relacional de objetos, aunque algunos programadores prefieren crear sus propias herramientas ORM.

**GRASP**: En diseño orientado a objetos, GRASP son patrones generales de software para asignación de responsabilidades, es el acrónimo de "GRASP (object-oriented design)|General Responsibility Assignment Software Patterns". Aunque se considera que más que patrones propiamente dichos, son una serie de "buenas prácticas" de aplicación recomendable en el diseño de software.

**S.O.L.I.D.**

•S: Single Responsibility Principle (SRP) - Principio de Responsabilidad Única

•O: Open-Closed Principle (OCP) – Principio de Abierto-Cerrado

•L: Liskov Substitution Principle (LSP) – Principio de Substitución de Liskov

•I: Interface Segregation Principle (ISP) – Principio de Segregación de Interface

•D: Dependency Inversion Principle (DIP) – Principio de Inversión de Dependencia

**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.

**OCJP:** antes El SCJP, Sun Certified Java Programmer, es el primer examen del programa de certificación. Es necesario para poder obtener las siguientes certificaciones y cubre los aspectos más básicos de la plataforma J2SE.

**SKYDRIVE:** SkyDrive permite a los usuarios subir archivos de una computadora y almacenarlos en línea (nube), y acceder a ellos desde un navegador web. El servicio utiliza Windows Live ID para controlar el acceso a los archivos del usuario, y les permite mantener la confidencialidad de los archivos, compartir con contactos o compartirlos con el público en general. Los archivos que se comparten públicamente no requieren una cuenta de Windows Live ID para acceder.

**DROPBOX:** Dropbox es un servicio de alojamiento de archivos multiplataforma en la nube, operado por la compañía Dropbox. El servicio permite a los usuarios almacenar y sincronizar archivos en línea y entre computadoras y compartir archivos y carpetas con otros.1 Existen versiones gratuitas y de pago, cada una de las cuales con opciones variadas

**JDK:** Java Development Kit o (JDK), es un software que provee herramientas de desarrollo para la creación de programas en Java. Puede instalarse en una computadora local o en una unidad de red.

**JRE:** Java Runtime Environment o JRE es un conjunto de utilidades que permite la ejecución de programas Java.

**GIT:** 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. 2 Sin embargo, Git se ha convertido desde entonces en un sistema de control de versiones con funcionalidad plena. 3 Hay algunos proyectos de mucha relevancia que ya usan Git, en particular, el grupo de programación del núcleo Linux.

**RCS:** Revision Control System o RCS es una implementación en software del control de versiones que automatiza las tareas de guardar, recuperar, registrar, identificar y mezclar versiones de archivos. RCS es útil para archivos que son modificados frecuentemente, por ejemplo programas informáticos, documentación, gráficos de procedimientos, monografías y cartas. RCS también puede ser utilizado para manejar archivos binarios, pero con eficacia y eficiencia reducidas. Las distintas versiones son archivadas mediante la ayuda de la herramienta diff.

**CVS:** Concurrent Versions System o simplemente CVS, también conocido como Concurrent Versioning System, es una aplicación informática que implementa un sistema de control de versiones: mantiene el registro de todo el trabajo y los cambios en los ficheros (código fuente principalmente) que forman un proyecto (de programa) y permite que distintos desarrolladores (potencialmente situados a gran distancia) colaboren. CVS se ha hecho popular en el mundo del software libre. Sus desarrolladores difunden el sistema bajo la licencia GPL.

**SVN**: Subversion es un sistema de control de versiones diseñado específicamente para reemplazar al popular CVS. Es software libre bajo una licencia de tipo Apache/BSD y se le conoce también como svn por ser el nombre de la herramienta utilizada en la línea de comando.

**MERCURIAL:** Mercurial es un sistema de control de versiones multiplataforma, para desarrolladores de software. Está implementado principalmente haciendo uso del lenguaje de programación Python, pero incluye una implementación binaria de diff escrita en C. Mercurial fue escrito originalmente para funcionar sobre Linux. Ha sido adaptado para Windows, Mac OS X y la mayoría de otros sistemas tipo Unix. Mercurial es, sobre todo, un programa para la línea de comandos. Todas las operaciones de Mercurial se invocan como opciones dadas a su programa motor, hg, que hace referencia al símbolo químico del mercurio.

**BAZAAR:** Bazaar es un sistema de control de versiones distribuido patrocinado por Canonical Ltd., diseñado para facilitar la contribución en proyectos de software libre y opensource.

Bazaar puede ser usado por un usuario único trabajando en múltiples ramas de un contenido local, o por un equipo colaborando a través de la red.

**MAQUINA VIRTUAL:** En informática una máquina virtual es un software que emula a una computadora y puede ejecutar programas como si fuese una computadora real. Este software en un principio fue definido como "un duplicado eficiente y aislado de una máquina física". La acepción del término actualmente incluye a máquinas virtuales que no tienen ninguna equivalencia directa con ningún hardware real.

**EC2:** Amazon EC2, es un servicio web ofrecido por Amazon. Proporciona herramientas de computación en nube de una manera flexible. Permite escalar los distintos servicios y necesidades de procesamiento de una manera fácil y ágil.

Posee una interfaz de servicios web para iniciar y configurar el servicio. Proporciona un control completo de los recursos y reduce el tiempo de arranque de sus servidores, lo que permite escalar recursos rápidamente según las necesidades del usuario. Provee herramientas de recuperación de datos y fuerte aislamiento frente a otros procesos realizados en sus máquinas.

**PARADIGMAS:** Un paradigma de programación es una propuesta tecnológica que es adoptada por una comunidad de programadores cuyo núcleo central es incuestionable en cuanto a que unívocamente trata de resolver uno o varios problemas claramente delimitados. La resolución de estos problemas debe suponer consecuentemente un avance significativo en al menos un parámetro que afecte a la ingeniería de software. Tiene una estrecha relación con la formalización de determinados lenguajes en su momento de definición. Un paradigma de programación está delimitado en el tiempo en cuanto a aceptación y uso ya que nuevos paradigmas aportan nuevas o mejores soluciones que la sustituyen parcial o totalmente.

**FUNCIONAL** es un paradigma de programación declarativa basado en la utilización de funciones aritméticas que no maneja datos mutables o de estado. Enfatiza la aplicación de funciones, en contraste con el estilo de programación imperativa, que enfatiza los cambios de estado. La programación funcional tiene sus raíces en el cálculo lambda, un sistema formal desarrollado en los 1930s para investigar la definición de función, la aplicación de las funciones y la recursión. Muchos lenguajes de programación funcionales pueden ser vistos como elaboraciones del cálculo lambda.

**LOGICO:** La programación lógica es un tipo de paradigmas de programación dentro del paradigma de programación declarativa. El resto de los subparadigmas de programación dentro de la programación declarativa son: programación funcional, programación con restricciones, programas DSL (de dominio específico) e híbridos. La programación lógica gira en torno al concepto de predicado, o relación entre elementos. La programación funcional se basa en el concepto de función (que no es más que una evolución de los predicados), de corte más matemático.

**ORIENTADA A OBJETOS:** La programación orientada a objetos o POO (OOP según sus siglas en inglés) es un paradigma de programación que usa los objetos en sus interacciones, para diseñar aplicaciones y programas informáticos. Está basado en varias técnicas, incluyendo herencia, abstracción, polimorfismo y encapsulamiento. Su uso se popularizó a principios de la década de los años 1990. En la actualidad, existe variedad de lenguajes de programación que soportan la orientación a objetos.





