



MINISTERIO
DE EDUCACIÓN, CULTURA
Y DEPORTE



FONDO SOCIAL EUROPEO
El FSE invierte en tu futuro

SECRETARÍA DE ESTADO DE
EDUCACIÓN, FORMACIÓN
PROFESIONAL Y
UNIVERSIDADES

DIRECCIÓN GENERAL
DE FORMACIÓN PROFESIONAL

INSTITUTO NACIONAL
DE LAS CUALIFICACIONES

GLOSARIO DE TÉRMINOS

CUALIFICACIÓN PROFESIONAL: PROGRAMACIÓN DE SISTEMAS INFORMÁTICOS

Código: IFC303_3

NIVEL: 3

4GL: Lenguaje de cuarta generación. Es un tipo de lenguaje de programación avanzado, en el que el estilo de programación es declarativo, en contraposición a imperativo o procedimental propio de los lenguajes de tercera generación como C, Java, etc. En este estilo, el lenguaje no indica qué acciones deben ejecutarse sino una serie de condiciones y es el propio compilador el que obtiene las instrucciones más adecuadas para cumplir tales condiciones. Son lenguajes más fáciles de usar y los entornos de programación suelen ser gráficos e incluyen capacidades de gestión avanzadas. Sin embargo, en algunos casos consumen muchos recursos y el código generado no está optimizado.

Análisis: Fase del ciclo de vida de desarrollo software que consiste en la identificación de los elementos, estructura, funcionalidades, relaciones, etc. de los elementos que se quieren desarrollar.

Ancho de banda: Es la cantidad de información que puede intercambiarse mediante una conexión de red en un período dado. El ancho de banda se indica generalmente en bits por segundo (bps), kilobits por segundo (Kbps), o megabits por segundo (Mbps).

Arquitectura de sistemas: Es una descripción del diseño y estructura de un sistema informático, tanto desde el punto de vista hardware como software.

Arquitectura de software: Es un conjunto de patrones que definen la estructura de un sistema software, es decir, los elementos que componen el sistema, sus propiedades y sus relaciones.

Base de datos: Es una colección de datos, organizados y almacenados para una fácil recuperación de estos.

Sistema Gestor de Bases de Datos (SGBD): Conjunto de programas y herramientas software que permiten la modificación, almacenamiento y extracción de los datos de una base de datos, así como su análisis. Estas herramientas aseguran la integridad, confidencialidad y seguridad de los datos.

Ciclo de vida software: Proceso que define las fases por las que pasa un proyecto de desarrollo de software. Existen diversos modelos del ciclo de vida software que determinan el orden de las fases del proceso y los criterios de transición entre fases, como por ej. En cascada, incremental, etc.

Clase: Definición de la estructura y comportamiento de un objeto en un enfoque orientado a objetos.

Cliente/Servidor: Arquitectura de sistemas que distingue dos partes: el servidor, encargado de ofrecer servicios tales como datos, documentos, aplicaciones y el cliente, que es quien realiza peticiones al servidor para obtener dichos servicios. Generalmente cliente y servidor se encuentran en máquinas (físicas o lógicas) independientes conectadas a través de una red de comunicaciones. En una

arquitectura cliente servidor, el proceso y eventualmente, parte de los datos, está distribuido entre los clientes y los servidores. A groso modo, los servidores almacenan información y la entregan a los clientes, que la solicitan.

Código ejecutable o código objeto: Código generado por un compilador o un ensamblador a partir de un código fuente de un programa y que generalmente puede ser ejecutado directamente por el procesador de un sistema informático.

Código fuente: Es el texto que contiene las instrucciones del programa, escritas en el lenguaje de programación. Se trata de un fichero de texto legible que se puede copiar, modificar e imprimir sin dificultad.

Componente: Es un recurso de software reusable, independiente y con un interfaz definido según unas especificaciones, que puede ser combinado con otros componentes ubicados en el mismo equipo u en otros equipos distribuidos en una red, para formar aplicaciones.

Comunicaciones asíncronas: En el ámbito informático es un tipo de comunicación que se establece entre dos o más sistemas informáticos de manera diferida en el tiempo, es decir, cuando no existe coincidencia temporal entre el emisor y receptor.

Comunicaciones síncronas: En el ámbito informático es un tipo de comunicación que se establece entre dos o más sistemas informáticos de manera que existe coincidencia temporal entre el emisor y receptor.

Controlador o manejador de dispositivo (driver): Es una aplicación informática integrada en un sistema operativo que permite interactuar con un dispositivo interno o externo al sistema informático. Este software da una capa de abstracción al hardware sobre el que se ha desarrollado.

Depuración de software (Debugging): Consiste en la revisión de una aplicación informática compilada con el fin de eliminar sus posibles errores y para conseguir que su funcionalidad y velocidad sean las máximas. Esta operación se realiza habitualmente mediante programas especializados denominados depuradores (debuggers).

Depurador (Debugger): Programa diseñado para ayudar a programar sin errores una aplicación, ayudando a identificar el error a corregir. Se pueden visualizar los valores de las variables en cada momento, establecer puntos de ruptura (instrucción que se marca indicando donde detener la ejecución del programa para ver hasta ese momento el resultado de la ejecución) y así subsanar errores lógicos.

Despliegue de software (software deployment): Es el conjunto de todas las actividades que permiten que un sistema de software pase a producción. El proceso general de despliegue de software consiste en diversas actividades

interrelacionadas y con transiciones entre ellas. Estas actividades pueden ocurrir tanto en el lado del desarrollador como el del cliente del sistema de software desarrollado. Generalmente el proceso está regulado mediante unas reglas, denominadas “normas de despliegue” que garantizan que el proceso se realiza con las debidas garantías de no interrupción del servicio, seguridad, recuperación o vuelta atrás, etc.

Diagrama de diseño: Representación gráfica de la solución ideada para el desarrollo de un proyecto software mediante la utilización de signos convencionales y utilizando generalmente un lenguaje estándar de modelización.

Diseño: Fase del ciclo de vida de desarrollo software que consiste en la definición de los recursos software que van a ser utilizados para desarrollar una aplicación conforme a los requisitos indicados en unas especificaciones dadas.

Dispositivo de Almacenamiento: Componente hardware que se utiliza para grabar y leer datos. Este dispositivo conforma la memoria o almacenamiento secundario de un equipo informático ~~ordenador~~. Pueden ser los conocidos discos duros, unidades de CD u otros tipos de dispositivos de almacenamiento.

Especificaciones: Conjunto de requisitos que deben ser cumplidos por un sistema software, tanto desde el punto de vista funcional como técnico.

Hardware: Conjunto de componentes materiales de un sistema informático o digital.

Implementación: Fase del ciclo de vida de desarrollo software que consiste en la codificación mediante lenguajes de programación de los algoritmos y estructuras de datos u objetos del sistema previamente definido en la fase de diseño.

Interfaz de componente: Es la definición de un conjunto de uno o más puntos de comunicación de entrada y/o salida de un componente, que permite su integración con otros componentes. Los interfaces de un componente permiten su utilización independientemente de cómo sea implementado.

Interfaz de usuario: Es el medio que permite a un usuario de un sistema informático comunicarse con el mismo. Pueden existir interfaces de usuario de diferentes tipos (gráficas, textuales, táctiles, gestuales, etc).

Lenguaje de programación: Es un lenguaje artificial que puede ser interpretado por un sistema informático mediante un programa compilador o un intérprete para su traducción a instrucciones del procesador (lenguaje máquina), el cual se utiliza para el desarrollo de software.

Lenguaje estructurado: Lenguaje de programación diseñado según el enfoque de programación estructurada.

Lenguaje de manipulación de datos: En el contexto de las bases de datos, un lenguaje de manipulación de datos es aquél específicamente diseñado para las funciones de manipulación de datos tales como la inserción, modificación y borrado de datos. Se utiliza en bases de datos de almacenamiento estructurado.

Librerías (bibliotecas) de software: Es un conjunto de programas agrupados que pueden ser utilizados por otros programas para el desarrollo de software. La composición de un programa a partir de funciones contenidas en librerías de software se denomina enlace (linking). Los sistemas operativos ofrecen habitualmente librerías que contienen los servicios básicos del sistema y que pueden ser utilizados por las aplicaciones para realizar sistemas de gestión y herramientas del sistema operativo. El término procede de una incorrecta traducción de la palabra en inglés *library*.

Monitorización: Proceso por medio del cual, nos aseguramos que nuestro proceder está encaminado adecuadamente hacia un resultado final, evitando las posibles desviaciones que pudieran presentarse.

Objeto: Entidad que encapsula unas propiedades o atributos y las operaciones que pueden realizarse sobre éstas garantizando que el cambio del estado de las propiedades sólo puede realizarse mediante las operaciones del objeto. En un enfoque orientado a objetos, generalmente los objetos se obtienen mediante la instanciación de clases.

Orientación a objetos: Es una forma de enfocar el desarrollo de software basada en la identificación de los datos y las operaciones encapsulados en una entidad denominada objeto. Los objetos instancian las clases, que basándose en sus propiedades a su vez pueden heredar características de otras clases (herencia) y redefinir sus atributos u operaciones (polimorfismo), etc.

Paquete de instalación: Es una agrupación de recursos software (programas, librerías, ficheros de configuración, datos, etc.) empaquetados generalmente en un único fichero comprimido para su distribución y despliegue. Los sistemas operativos cuentan generalmente con sistemas de gestión de paquetes conformes a un determinado formato y que facilitan la instalación, desinstalación y actualización de los paquetes. Los paquetes habitualmente pueden obtenerse de repositorios públicos gestionados por alguna entidad o comunidad de usuarios.

Periféricos: Son los dispositivos hardware independientes conectados a un sistema informático. Ejemplos de periféricos son los dispositivos de entrada y salida (pantalla, teclado, ratón, etc.), de almacenamiento (unidades de backup, lectores ópticos, unidades USB), de comunicaciones (modem, adaptador wifi), etc

Proceso: Es un programa en ejecución. Los procesos son gestionados mediante el controlador de procesos del sistema operativo. Básicamente un proceso tiene

un estado de ejecución, una memoria de trabajo y eventualmente un conjunto de procesos “hijo” dependientes de él. Los procesos también suelen tener asociada una lista de ficheros abiertos vinculados a su ejecución. El controlador de procesos se encarga de repartir la carga de memoria y consumo de procesador de los procesos para que su ejecución sea lo más fluida posible y no se produzcan bloqueos del sistema informático en su conjunto.

Programación: Es el proceso de desarrollar programas informáticos, es decir, crear, probar, depurar y mantener el código fuente de las aplicaciones informáticas. Generalmente implica el diseño de algoritmos y estructuras de datos (comúnmente agrupados en clases cuando se utiliza un enfoque orientado a objetos) y la codificación en uno o varios lenguajes de programación.

Programación estructurada: Es un estilo de programación que utiliza estructuras sencillas tales como secuencias, condicionales, bucles, etc. Surgió como una alternativa a los problemas de mantenimiento de código producidos por una programación mediante saltos (goto) a cualquier punto del programa que derivó en aplicaciones conocidas como “espaguetti”. La evolución de la programación estructurada fue la programación orientada a objetos, mediante la cual, aparte de obtenerse programas estructurados también se consiguen abstraer datos y operaciones en una única entidad, modular y reutilizable.

Protocolo (de comunicaciones): Conjunto de reglas que permiten la comunicación entre varios sistemas informáticos a través de líneas de comunicaciones. Los protocolos pueden ser realizados mediante hardware, software o una combinación de ambos. Habitualmente los protocolos forman parte de una arquitectura en capas mediante la cual un protocolo superior hace uso de otros protocolos de nivel inferior para realizar sus funciones. Ejemplos de este tipo de arquitecturas son por ejemplo la pila de protocolos TCP/IP o el clásico OSI de ISO, donde ciertos protocolos se usan en las transferencias realizadas sobre LANs físicas (ApleTalk y PPP, entre otros) y otros como FTP y HTTP se encapsulan sobre los primeros para completar la distribución de contenidos multimedia, correo electrónico, ficheros de datos, etc.

Pruebas de software: Proceso que consiste en la comprobación del software desarrollado en la fase de implementación no contiene errores. Este objetivo se consigue generalmente mediante la definición y realización de pruebas utilizando herramientas de permiten la automatización de las mismas.

Red de Área Local (LAN): Una red de área local o LAN (Local Area Network) es la interconexión de varias computadoras y periféricos entre si. Su extensión está limitada a un entorno cercano. Su aplicación más extendida es la interconexión de computadoras personales y estaciones de trabajo en oficinas, o fábricas, por ejemplo. El término red local incluye tanto el hardware como el software necesario para la interconexión de los distintos dispositivos y el tratamiento de la información, de forma que permite a sus usuarios el intercambio de datos y la compartición de recursos.

Reusabilidad: Posibilidad de que un elemento de software pueda ser usado nuevamente sin necesidad de ser reimplementado, simplificando y agilizando el desarrollo de aplicaciones.

Servidor: Es una computadora que, formando parte de una red, provee servicios a otras computadoras.

Servidor LDAP: Es un servidor de datos orientado para almacenar usuarios y permitir consultar si es correcto o no una tupla usuario y contraseña.

Sistema operativo: Conjunto de programas que controlan el funcionamiento de un ordenador. Ejemplos de sistemas operativos son Windows, GNU/Linux, Unix, IOS, etc. El sistema operativo es el encargado de gestionar los elementos de un sistema informático (procesos, memoria, almacenamiento, etc.) así como los periféricos conectados a él (impresoras, escáneres, dispositivos de comunicaciones, etc.).

Sistemas de ficheros: Son los componentes que estructuran la información guardada en una unidad de almacenamiento (normalmente un disco duro de una computadora). La mayoría de los sistemas operativos manejan sus propios sistemas de ficheros.

Sistemas de Monitorización: En el contexto de la ingeniería del software son sistemas capaces de obtener información de un sistema software para su posterior análisis. Este tipo de sistemas facilita la detección de problemas de rendimiento, seguridad, etc. de sistemas software en producción.

Sniffer o analizador de red: Es una herramienta que sirve para examinar en profundidad el tráfico de red, depurar protocolos y aplicaciones de red. Permite al ordenador capturar diversas tramas de red para analizarlas, ya sea en tiempo real o después de haberlas capturado. Por analizar se entiende que el programa puede reconocer que la trama capturada pertenece a un protocolo concreto (TCP, ICMP...) y muestra al usuario la información decodificada. De esta forma, el usuario puede ver todo aquello que en un momento concreto está circulando por la red que se está analizando.

Software: Conjunto de componentes no materiales de un sistema informático o digital. El término inglés original define el concepto por oposición a hardware en referencia a la intangibilidad de los programas.

SSL/TLS: Secure Sockets Layer -Protocolo de Capa de Conexión Segura- (SSL) y Transport Layer Security -Seguridad de la Capa de Transporte- (TLS), que es el sucesor de SSL, son protocolos que permiten una comunicación segura entre los nodos de una red, generalmente pública como Internet, utilizando mecanismos criptográficos. Proporcionan integridad, privacidad y autenticación de los sistemas informáticos intervinientes en la comunicación. Esto es posible

por la utilización de claves asimétricas y simétricas de una longitud elevada (por ej. mayor de 128 bits) y algoritmos criptográficos avanzados como RSA o AES. Generalmente se utilizan como una capa de seguridad para protocolos no seguros como http (en cuyo caso se denomina https) o smtp.

TCP/IP: Familia de protocolos de Internet que toma su nombre de los dos protocolos más importantes: TCP (Protocolo de control de la transmisión) e IP (Protocolo de Internet). La familia completa de protocolos TCP/IP está formada de aproximadamente un centenar entre los que destacan HTTP (HyperText Transfer Protocol), para el acceso a páginas web, FTP (File Transfer Protocol) para transferencia de ficheros, POP (Post Office Protocol) para la recepción de correo electrónico, SMTP (Simple Mail Transfer Protocol) para el envío de correo, etc.

Tipos de datos: En el contexto de la programación de sistemas informáticos, un tipo de dato es un atributo asociado a un dato o variable que indica sus características y que facilita al compilador la codificación en memoria del mismo. Ejemplos de tipos de datos son: caracteres, enteros, fechas, etc. La utilización de tipos de datos en un lenguaje de programación ayuda a evitar errores, puesto que en tiempo de compilación es posible detectar si se intenta programar la manipulación de un dato mediante operaciones incorrectas, como por ej. multiplicar una fecha o sumar un carácter y un número. Generalmente los lenguajes de programación soportan tipos de datos predefinidos y permiten la definición de tipos de datos por el programador.

UML: Lenguaje unificado de modelización. Es un lenguaje de modelización de sistemas software ampliamente utilizado estandarizado por la organización OMG (Object Management Group). Se trata de un lenguaje gráfico y su uso se basa en la creación de diferentes diagramas que abarcan diferentes aspectos de un sistema software: estructura, actividades, estados, etc. Habitualmente se utiliza en las fases de análisis y diseño del ciclo de vida software, aunque también existe un enfoque denominado MDA, igualmente promovido por OMG cuyo objetivo es la implementación de sistemas software mediante la generación automática de código a partir de diagramas UML.

Usuario: Es la persona que utiliza o trabaja con algún objeto o que es destinataria de algún servicio público, privado, empresarial o profesional. En sentido general, un usuario es un conjunto de permisos y de recursos (o dispositivos) a los cuales se tiene acceso. Es decir, un usuario puede ser tanto una persona como una máquina, un programa, etc

Validación de software: Proceso mediante el cual se verifica que el sistema que se ha desarrollado en la fase de implementación cumple los requisitos establecidos en las especificaciones.