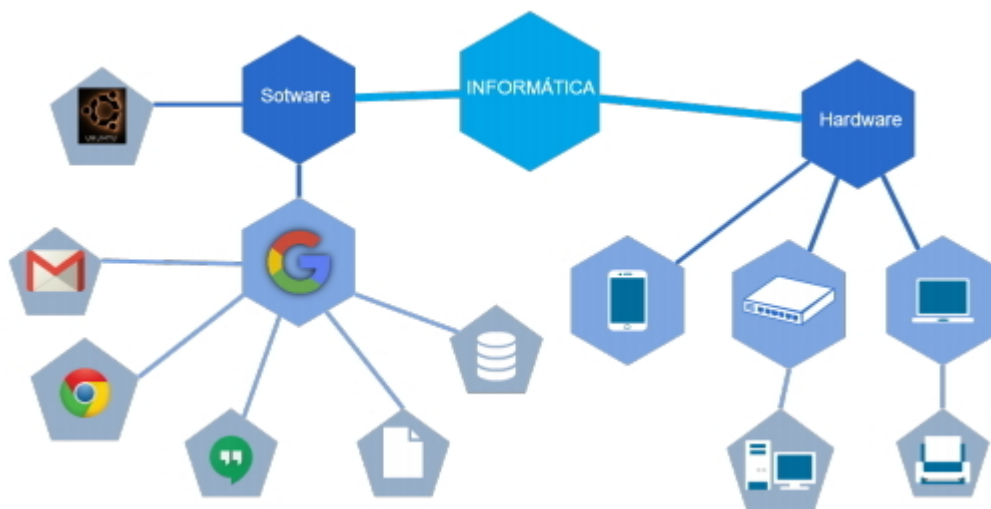


Conceptos iniciales de programación

Hardware y software

¿Qué es hardware y software?



Los dispositivos tecnológicos como computadoras o smartphones están compuestos por hardware y software.

Hardware es el conjunto de componentes físicos de los que está hecho el equipo y **software** es el conjunto de programas o aplicaciones, instrucciones y reglas informáticas que hacen posible el funcionamiento del equipo.

Lenguajes y frameworks

¿Qué es un lenguaje de programación?

Se conoce como lenguaje de programación a un programa destinado a la construcción de otros programas informáticos. Su nombre se debe a que comprende un lenguaje formal que está diseñado para organizar algoritmos y procesos lógicos que serán luego llevados a cabo

por un ordenador o sistema informático, permitiendo controlar así su comportamiento físico, lógico y su comunicación con el usuario humano.

Dicho lenguaje está compuesto por símbolos y reglas sintácticas y semánticas, expresadas en forma de instrucciones y relaciones lógicas, mediante las cuales se construye el código fuente de una aplicación o pieza de software determinado. Así, puede llamarse también lenguaje de programación al resultado final de estos procesos creativos.

La implementación de lenguajes de programación permite el trabajo conjunto y coordinado, a través de un conjunto de instrucciones posibles, de diversos programadores o arquitectos de software, para lo cual estos lenguajes imitan, al menos formalmente, la lógica de los lenguajes humanos o naturales.

Algunos ejemplos de lenguajes de programación son: C#, C++, Python, Java, Javascript, PHP, etc.

¿Qué es un Framework?

Un Framework, que se podría traducir aproximadamente como marco de trabajo, es el esquema o estructura que se establece para desarrollar y organizar un software determinado. Esta definición, algo compleja, podría resumirse como el entorno pensado para hacer más sencilla la programación de cualquier aplicación o herramienta actual.

Este sistema plantea varias ventajas para los programadores, ya que automatiza muchos procesos y además facilita el conjunto de la programación. Es útil, por ejemplo, para evitar el tener que repetir código para realizar funciones habituales en un rango de herramientas, como puede ser el acceder a bases de datos o realizar llamadas a Internet. Todas estas tareas son las que se realizan de forma mucho más fácil cuando se trabaja dentro de un framework.

Para qué sirve un Framework

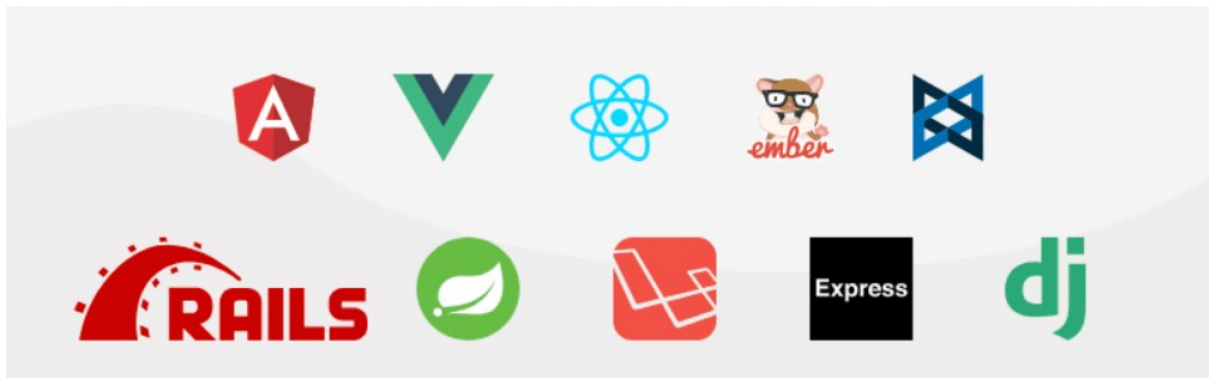
Un Framework sirve para poder escribir código o desarrollar una aplicación de manera más sencilla. Es algo que permite una mejor organización y control de todo el código elaborado, así como una posible reutilización en el futuro. Debido a esto, garantiza una mayor productividad que los métodos más convencionales y una minimización del coste al agilizar las horas de trabajo volcadas en el desarrollo.

Por otra parte, su acción es algo que afecta también a los errores, minimizándolos considerablemente. En definitiva, es algo que brinda una ayuda general y más que considerable al desarrollador, haciendo que sus labores sean mucho más sencillas.

Ejemplos de Framework

Existen numerosos frameworks en la actualidad. Dado que la cantidad de dispositivos se ha multiplicado en los últimos años, así como su forma de trabajar la información y los sistemas a los que acuden, es normal que hayan ido apareciendo más y más nombres destinados a hacer más fáciles las tareas de programación.

De esta forma, podemos encontrarnos con nombres muy representativos como .NET, Reactjs, Express, Sails, Angular, Vuejs .etc.



intérprete y compiladores

Compilador

Proceso de traducción que convierte un programa fuente escrito en un lenguaje de alto nivel a un programa objeto en código máquina y listo por tanto para ejecutarse en la computadora. A grandes rasgos un compilador es un programa que lee un programa escrito en un lenguaje, el lenguaje fuente, y lo traduce a un programa equivalente en otro lenguaje, el lenguaje objeto. Como parte importante de este proceso de traducción, el compilador informa a su usuario de la presencia de errores en el programa fuente.

Intérprete

Un intérprete analiza el programa fuente y lo ejecuta directamente, sin generar ningún código equivalente. Es un traductor que realiza la operación de compilación paso a paso. Para cada sentencia que compone el texto de entrada, se realiza una traducción, ejecuta dicha sentencia y vuelve a iniciar el proceso con la sentencia siguiente.

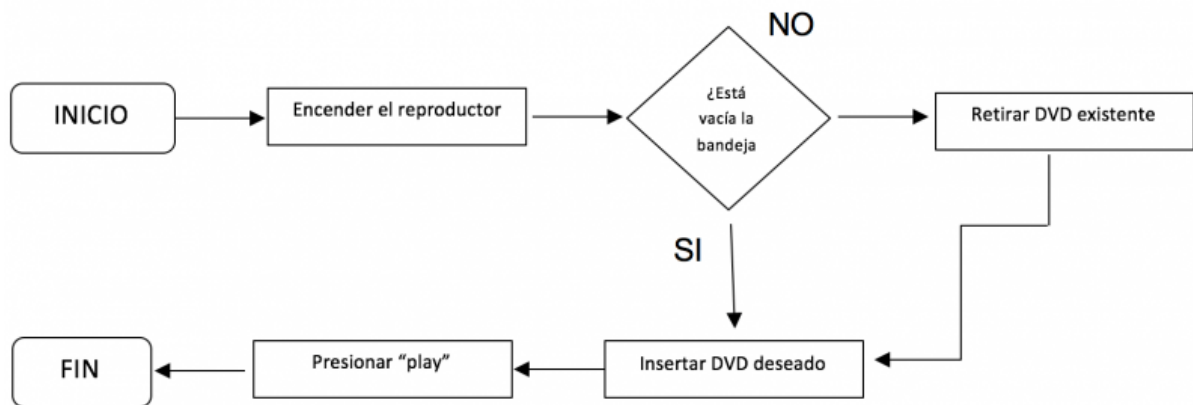
La principal ventaja del proceso de compilación frente al de interpretación es que los programas se ejecutan mucho más rápidamente una vez compilados; por el contrario, es más cómodo desarrollar un programa mediante un intérprete que mediante un compilador puesto que en el intérprete las fases de edición y ejecución están más integradas. La depuración de los programas suele ser más fácil en los intérpretes que en los compiladores puesto que el código fuente está presente durante la ejecución. Estas ventajas pueden incorporarse al compilador mediante la utilización de entornos de desarrollo y depuradores simbólicos en tiempo de ejecución.

Diagramas de Flujo

¿Que es un diagrama de flujo?

El diagrama de flujo, se encarga de representar una secuencia de pasos lógicos, necesarios para realizar una tarea. Es una representación gráfica donde se desglosa de forma secuencial un proceso, se usan en cualquier actividad dentro de las empresas industriales o de servicios.



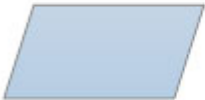


Se trata de una herramienta que facilita el entendimiento de cada una de las fases de un proceso y su funcionamiento, permitiendo por lo tanto la posibilidad de estudiarlo para mejorar sus procedimientos.



Cómo hacer un diagrama de flujo

Los diagramas de flujo usan formas especiales para representar diferentes tipos de acciones o pasos en un proceso. Las líneas y flechas muestran la secuencia de los pasos y las relaciones entre ellos. Todos estas formas especiales son conocidas como símbolos de diagrama de flujo.

El tipo de diagrama dicta los símbolos de diagramas de flujo que se utilizan. Por ejemplo, un diagrama de flujo de datos puede contener un Símbolo de Entrada o Salida (también conocido como Símbolo de E/S).

Símbolo	Nombre	Función
	Inicio / Final	Representa el inicio y el final de un proceso
	Linea de Flujo	Indica el orden de la ejecución de las operaciones. La flecha indica la siguiente instrucción.
	Entrada / Salida	Representa la lectura de datos en la entrada y la impresión de datos en la salida
	Proceso	Representa cualquier tipo de operación
	Decisión	Nos permite analizar una situación, con base en los valores verdadero y falso

Pasos para construir el diagrama de flujo

1. Definir el propósito y el alcance del problema a resolver.
2. Identificar las tareas necesarias
3. Organizar las tareas con las figuras antes mencionadas es decir, procesos, decisiones, tareas, datos, entradas y/o salidas. Situándonos en el orden pertinente en que cada actividad es ejecutada dentro del proceso.
4. Creación del diagrama, al tener clara la idea, se plasma un borrador y se dibuja a mano o usando algún programa para diseño de diagramas de flujo

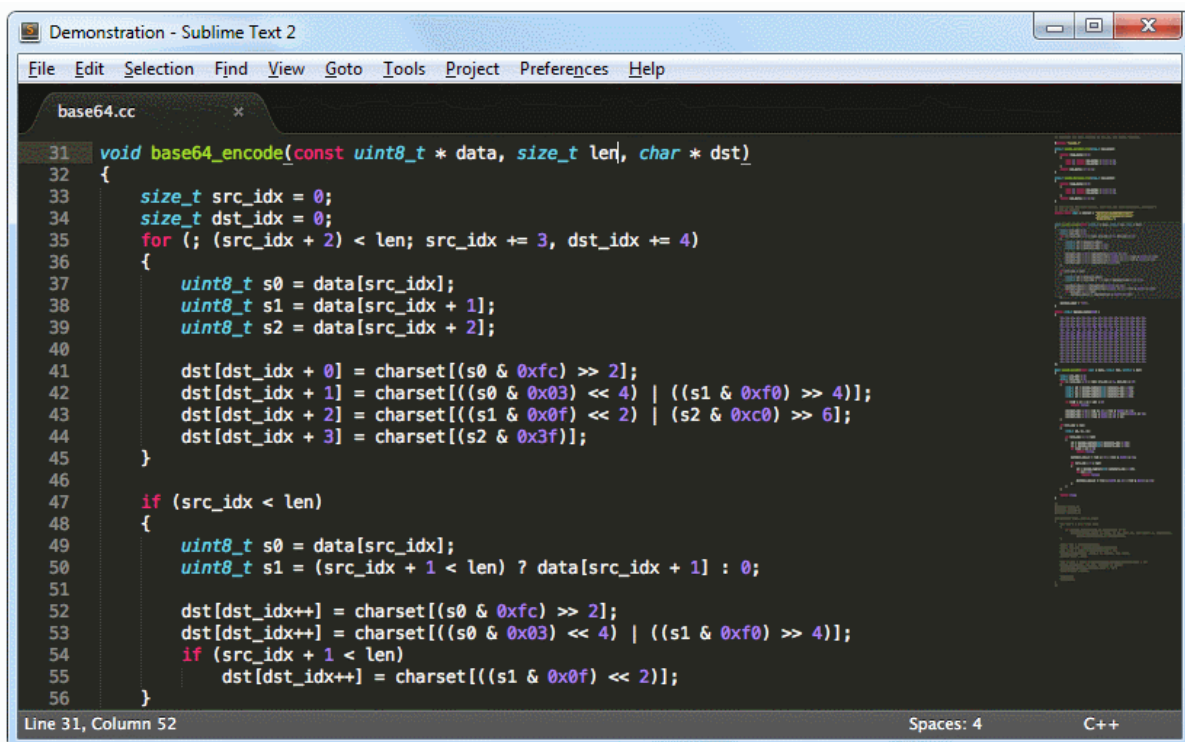
La construcción de este tipo de diagramas se utiliza para poder comprender y analizar mejor, los pasos que debemos realizar en un programa para encontrar la solución que se desea mostrar.

Editor de Texto

Dentro de la categoría de editores de programación no existe una elección correcta, sino que hay disponibles muchas opciones y debemos elegir la que más nos guste personalmente. Algunas personas se inclinan por editores de texto minimalistas, que reduzcan las distracciones, mientras que otras personas prefieren utilizar un IDE (entorno de desarrollo integrado), donde prima la automatización de tareas, depuración y la especialización a ciertos lenguajes de programación específicos.

Sublime Text

Sublime Text es uno de los mejores editores que existen actualmente. Destaca por lo ligero y simple que es, el aspecto visual sencillo pero elegante, los automatismos para realizar cambios de código, pero sobre todo por su gran capacidad de personalización: Sublime Text incorpora una gran cantidad de plugins para hacer prácticamente cualquier cosa.



```
31 void base64_encode(const uint8_t * data, size_t len, char * dst)
32 {
33     size_t src_idx = 0;
34     size_t dst_idx = 0;
35     for (; (src_idx + 2) < len; src_idx += 3, dst_idx += 4)
36     {
37         uint8_t s0 = data[src_idx];
38         uint8_t s1 = data[src_idx + 1];
39         uint8_t s2 = data[src_idx + 2];
40
41         dst[dst_idx + 0] = charset[(s0 & 0xfc) >> 2];
42         dst[dst_idx + 1] = charset[((s0 & 0x03) << 4) | ((s1 & 0xf0) >> 4)];
43         dst[dst_idx + 2] = charset[((s1 & 0x0f) << 2) | (s2 & 0xc0) >> 6];
44         dst[dst_idx + 3] = charset[(s2 & 0x3f)];
45     }
46
47     if (src_idx < len)
48     {
49         uint8_t s0 = data[src_idx];
50         uint8_t s1 = (src_idx + 1 < len) ? data[src_idx + 1] : 0;
51
52         dst[dst_idx++] = charset[(s0 & 0xfc) >> 2];
53         dst[dst_idx++] = charset[((s0 & 0x03) << 4) | ((s1 & 0xf0) >> 4)];
54         if (src_idx + 1 < len)
55             dst[dst_idx++] = charset[((s1 & 0x0f) << 2)];
56     }
```

Está disponible para todas las plataformas y, aunque se trata de un producto comercial, puede probarse de forma gratuita su versión completa, sin funcionalidades limitadas.

Web de Sublime: <https://www.sublimetext.com/>

Visual Code

VS Code es un editor de código fuente sofisticado que admite muchas funcionalidades prácticas al momento de trabajar con el código, pertenece a la empresa Microsoft y es open source. Admite diferentes lenguajes de programación como: Java, Go, C, C++, Ruby, Python, PHP, Perl, JavaScript, Groovy, Swift, PowerShell, Rust, DockerFile, CSS, HTML, XML, JSON, Lua, F#, Batch, SQL, Objective-C...

Al igual que Sublime es multiplataforma, es decir, funciona en Windows, Linux y Mac OS. También puede ser optimizado con numerosos plugins que existen.

VS, posee «Intellisense», esta es la capacidad que tiene un editor de texto para predecir la instrucción que estamos por escribir, y con esto no tenemos la necesidad de escribir toda la instrucción, ya que esta se puede autocompletar con el editor, esto nos hace más productivos y acorta la posibilidad de errores de sintaxis.

