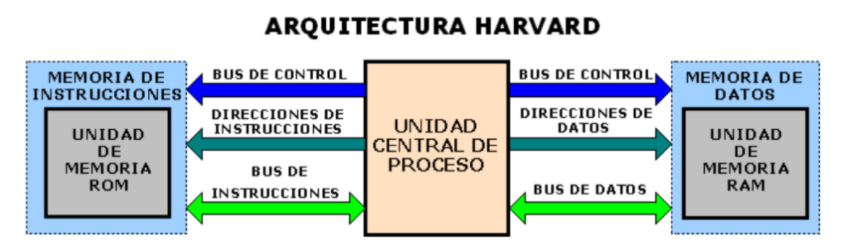
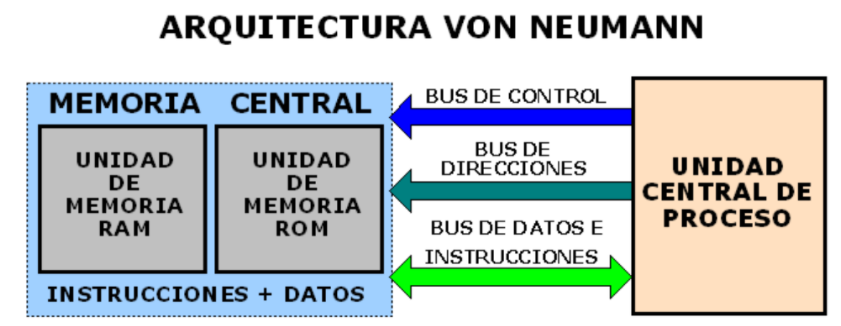
DICCIONARIO

* **Transistor**: El transistor es un dispositivo electrónico semiconductor utilizado para entregar una señal de salida en respuesta a una señal de entrada.
* **Ley-de-moore:** expresa que aproximadamente cada 2 años se duplica el número de [transistores](https://es.wikipedia.org/wiki/Transistor) en un [microprocesador](https://es.wikipedia.org/wiki/Microprocesador).
* **Puertas-lógicas:** Una puerta lógica, o compuerta lógica, es un dispositivo electrónico con una función booleana u otras funciones como sumar o restar, incluyen o excluyen según sus propiedades lógicas. XOR / AND / OR/ NOR/
* **Periférico:** En informática, periférico es la denominación genérica para designar al aparato o dispositivo auxiliar e independiente conectado a la placa base de una computadora.
* **Arquitectura-von-neumann:** La arquitectura de la ALU / CPU etc.
* **Arquitectura-harvard:** La diferencia principal entre las dos arquitecturas se encuentra en el mapa de memoria: mientras que en la **arquitectura Von Neumann** hay un único espacio de memoria para datos y para instrucciones, en la **arquitectura Harvard** hay dos espacios de memoria separados: un espacio de memoria para los datos y un espacio de instrucciones.





* **Sistema-operativo:** es un conjunto de programas que permite manejar la memoria, disco, medios de almacenamiento de información y los diferentes periféricos o recursos de nuestra computadora, como son el teclado, el mouse, la impresora, la placa de red, entre otros.
* **Hardware**: Conjunto de elementos físicos o materiales que constituyen una computadora o un sistema informático.
* **Software**: Se conoce como software​, logicial o soporte lógico al sistema formal de un sistema informático, que comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos que son llamados hardware.
* **Raspberry pi:** es un ordenador del tamaño de una tarjeta de crédito. ... Consiste en una placa base que soporta distintos componentes de un ordenador como un procesador ARM de hasta 1500 MHz, un chip gráfico y una memoria RAM de hasta 8 GB. Además, tiene otras muchas otras posibilidades.
* **Arduino**: El arduino es una placa que tiene todos los elementos necesarios para conectar periféricos a las entradas y salidas de un microcontrolador. Es decir, es una placa impresa con los componentes necesarios para que funcione el microcontrolador y su comunicación con un ordenador a través de la comunicación serial. **Arduino es una plataforma de creación de electrónica de código abierto**basada en hardware y software libre, lo que permite que cualquiera pueda utilizarlos y adaptarlos.

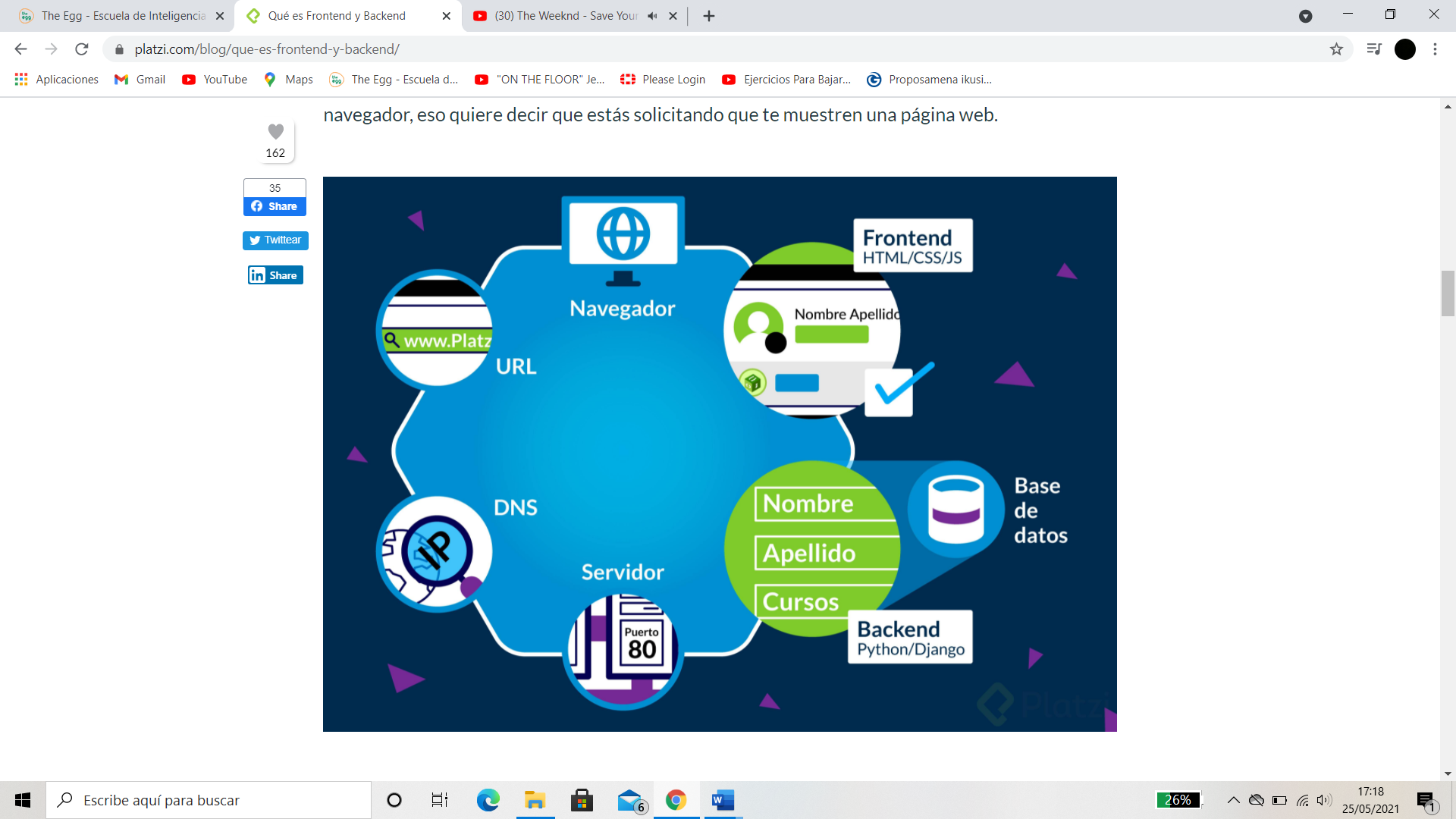
**Arduino y Raspberry Pi son dos conceptos totalmente diferentes**, por lo que es un poco difícil hacer una comparación entre ambos. Son dos productos con diferentes finalidades, aunque por su versatilidad la imaginación de la comunidad maker haya hecho que ambos sean utilizados para crear [todo tipo de proyectos de electrónica](https://www.xataka.com/makers/46-proyectos-makers-para-hacer-verano-arduino-raspberry-pi).

Aunque ambos son proyectos cuyo software es de código es abierto y cualquiera puede revisarlo, **una de las principales diferencias está en la filosofía del hardware**. El hardware de Arduino es también abierto para que cualquiera pueda crear sus propias versiones de la placa, mientras que la Raspberry Foundation mantiene el control sobre las placas Raspberry Pi, y sólo ellos las crean y fabrican.

* **Compilador:** es un tipo de traductor que transforma un programa entero de un lenguaje de programación a otro.​ Usualmente el lenguaje objetivo es código máquina, aunque también puede ser traducido a un código intermedio o a texto.

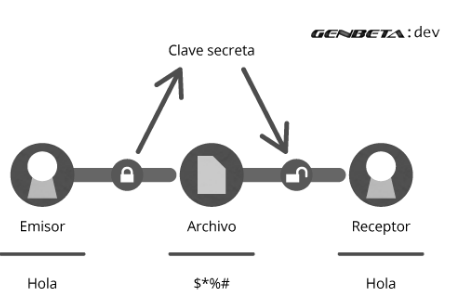
Proceso de traducción que convierte un programa fuente escrito en un lenguaje de alto nivel a un programa objeto en código máquina y listo por tanto para ejecutarse en la computadora.

* **Intérprete**: que analiza el programa fuente y lo ejecuta directamente, sin generar ningún código equivalente.
* **Lenguaje-de-programación:** es un [lenguaje formal](https://es.wikipedia.org/wiki/Lenguaje_formal) (o artificial, es decir, un lenguaje con reglas gramaticales bien definidas) que le proporciona a una persona, en este caso el programador, la capacidad de escribir (o programar) una serie de [instrucciones](https://es.wikipedia.org/wiki/Instrucci%C3%B3n_(inform%C3%A1tica)) o secuencias de órdenes en forma de [algoritmos](https://es.wikipedia.org/wiki/Algoritmo) con el fin de controlar el comportamiento físico o lógico de un [sistema informático](https://es.wikipedia.org/wiki/Sistema_inform%C3%A1tico), de manera que se puedan obtener diversas clases de datos o ejecutar determinadas tareas. A todo este conjunto de órdenes escritas mediante un lenguaje de programación se le denomina [programa informático](https://es.wikipedia.org/wiki/Software). Por ejemplo, Python, R, Java, Lisp.
* **Editor-de-código:** Un editor de código fuente es un editor de texto diseñado específicamente para editar el código fuente de programas informáticos. Puede ser una aplicación individual o estar incluido en un entorno de desarrollo integrado.
* **Lenguaje-de-alto-nivel:** Se procesa mucho más lento. nos referimos al tipo de lenguaje de programación que no expresa los algoritmos teniendo en cuenta la capacidad que tienen las máquinas para ejecutar órdenes, sino al que se utiliza teniendo en cuenta las capacidades cognitivas de los seres humanos.
* **Lenguaje de máquina**:  es el sistema de códigos directamente interpretable por un circuito microprogramable, como el microprocesador de una computadora o el microcontrolador de un autómata.
* **Frontend** es la parte de un sitio web que interactúa con los usuarios, por eso decimos que está del lado del cliente.
* **Backend** es la parte que se conecta con la base de datos y el servidor que utiliza dicho sitio web, por eso decimos que el backend corre del lado del servidor.

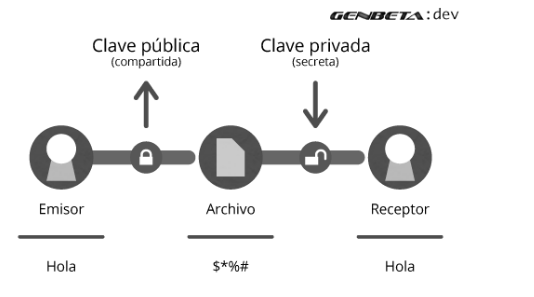


Si estás aprendiendo a programar, seguramente vas a encontrarte con muchos de los términos de este artículo. Además, en algún punto vas a tener que decidir si prefieres el [Desarrollo Frontend o el Desarrollo Backend](https://platzi.com/web), aunque es cierto que hay [Desarrolladores FullStack](https://www.youtube.com/watch?v=ePZBgZfoQLE) que involucran las dos áreas.

* **Repositorio de control de versiones:** Necesitamos una herramienta que gestione las partes del programa de cada individuo y todas sus versiones. Pero tranquilos, todo está inventado. Estas herramientas se conocen como repositorios de control de versiones y son una especie de biblioteca dónde todo se guarda de una manera organizada. Existen diferentes marcas: GitHub, bitbucket, subversion, etcétera.
* **Algoritmo:** Es un conjunto de instrucciones o reglas definidas y no-ambiguas, ordenadas y finitas que permite, solucionar un problema y llevar a cabo otras tareas o actividades.​
* **Diagrama de flujo:** Es la representación gráfica de un algoritmo o proceso. Permite ordenar ideas por los que a partir del mismo se escribe un programa en cualquier lenguaje de programación. Se utilizan tanto para planificar el desarrollo de un software como para diseños en otras disciplinas que nada tienen que ver con la informática.
* **Código RSA:**
* **Algoritmo de cifrado simétrico:** Este cifrado funciona con una sola clave. El emisor y el receptor deberá conocer la clave y el mensaje se cifrará y descifrará con la misma clave.



* **Algoritmo de cifrado asimétrico:** Este algoritmo funciona con dos claves, una clave pública (se puede difundir sin problema) y la otra privada (no se podrá revelar nunca). Las dos claves estarán vinculadas. Si queremos que tres compañeros de trabajo nos manden un archivo cifrado debemos de mandarle nuestra clave pública (que está vinculada a la privada) y nos podrán mandar de forma confidencial ese archivo que solo nosotros podremos descifrar con la clave privada.



La criptografía simétrica es más insegura ya que el hecho de pasar la clave es una gran vulnerabilidad, pero se puede cifrar y descifrar en menor tiempo del que tarda la criptografía asimétrica, que es el principal inconveniente y es la razón por la que existe la criptografía híbrida.

* **Criptografía híbrida:**
  + Generar una clave pública y otra privada (en el receptor).
  + Cifrar un archivo de forma síncrona.
  + El receptor nos envía su clave pública.
  + Ciframos la clave que hemos usado para encriptar el archivo con la clave pública del receptor.
  + Enviamos el archivo cifrado (síncronamente) y la clave del archivo cifrada (asíncronamente y solo puede ver el receptor).
* **Funciones en programación:**
* **Algoritmo Hash:** Algoritmo de huella. Es una función matemática que nos genera una huella. Es la huella de un archivo, esta huella es siempre única. Algoritmo de cifrado de un solo camino. Se usa para verificar la autenticidad del archivo. Hay varias versiones desde 0 hasta el 5.