

Tarea para DWEC01.

1. Instalación de navegadores web

Estamos en un entorno de Ubuntu 18.04, con lo que nuestro navegador predeterminado será Mozilla Firefox. Procederemos a instalar **Microsoft Edge**, **Google Chrome** y **Opera**. Esto nos permitirá cubrir el [90.05%](#) del mercado de navegadores web de escritorio en España. Como se puede observar en el siguiente gráfico, Safari tiene la 2ª mayor cuota de mercado en España, pero para instalarlo tendríamos que usar WINE, así que lo dejaremos de lado.

Como editor de páginas web vamos a instalar **Visual Studio Codium**. Esto es una variante de Visual Studio Code de Microsoft, donde usan el código fuente de Visual Studio con licencia MIT y le quitan los elementos de rastreo y telemetría que tiene la versión oficial de Microsoft.

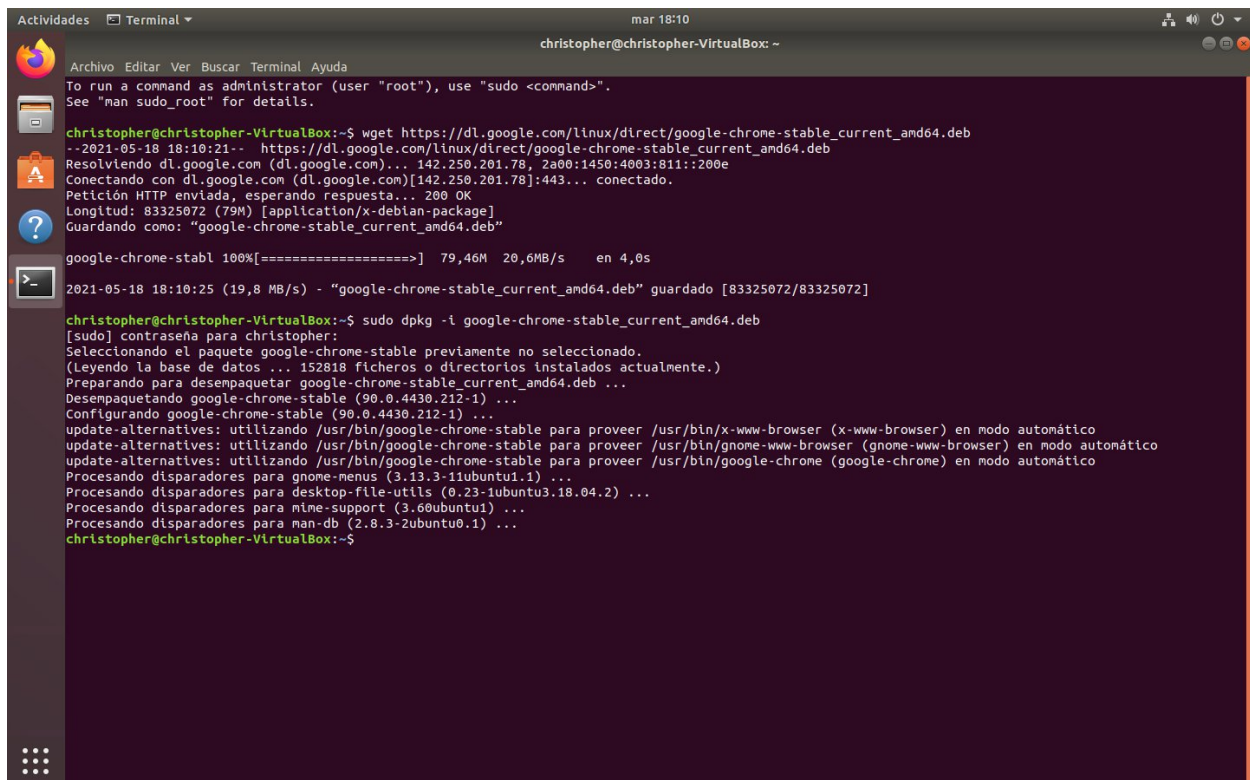
1.1 Instalación de Google Chrome.

Para instalar Google Chrome usaremos la terminal y emplearemos estos dos comandos:

o `wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb`

o `sudo dpkg -i google-chrome-stable_current_amd64.deb`

El primer comando nos descargará los archivos de instalación de Chrome, y el segundo lo instalará con el gestor de paquetes dpkg.

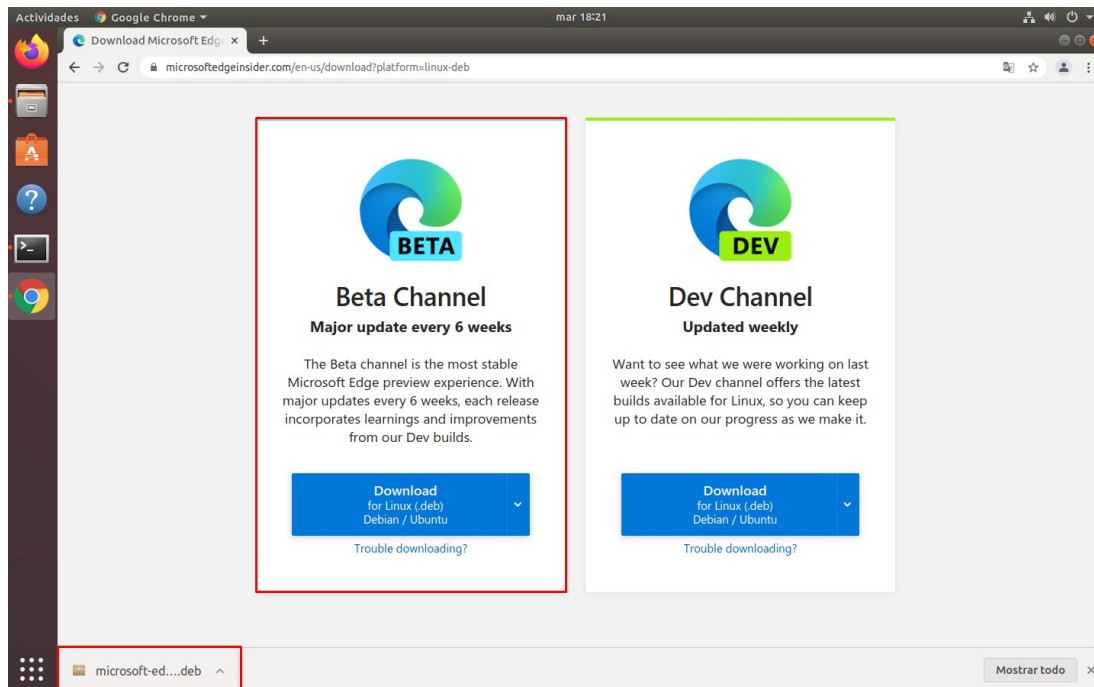


```
christopher@christopher-VirtualBox: ~
$ wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
--2021-05-18 18:10:21-- https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
Resolviendo dl.google.com (dl.google.com)... 142.250.201.78, 2a00:1450:4003:811::200e
Conectando con dl.google.com (dl.google.com)[142.250.201.78]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 83325072 (79M) [application/x-debian-package]
Guardando como: "google-chrome-stable_current_amd64.deb"
google-chrome-stabl 100%[=====] 79,46M 20,6MB/s en 4,0s
2021-05-18 18:10:25 (19,8 MB/s) - "google-chrome-stable_current_amd64.deb" guardado [83325072/83325072]

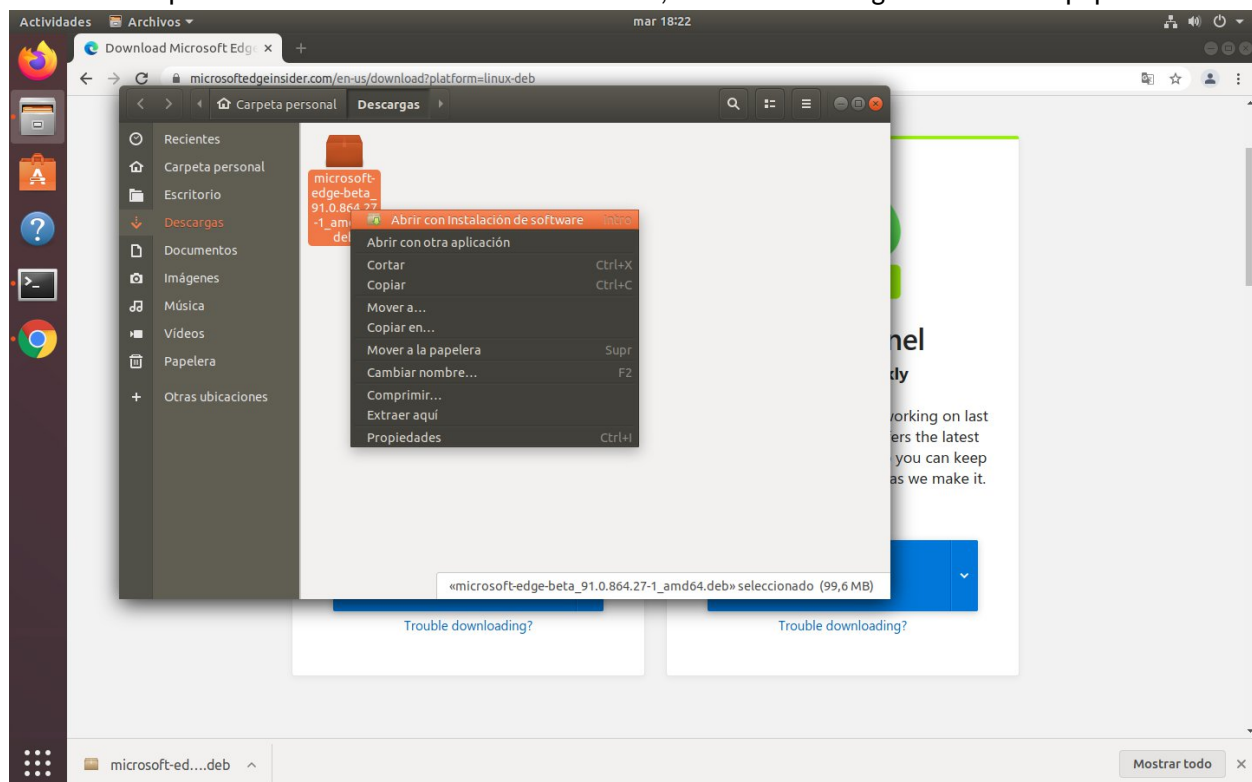
christopher@christopher-VirtualBox:~$ sudo dpkg -i google-chrome-stable_current_amd64.deb
[sudo] contraseña para christopher:
Seleccionando el paquete google-chrome-stable previamente no seleccionado.
(Leyendo la base de datos ... 152818 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar google-chrome-stable_current_amd64.deb ...
Desempaquetando google-chrome-stable (90.0.4430.212-1) ...
Configurando google-chrome-stable (90.0.4430.212-1) ...
update-alternatives: utilizando /usr/bin/google-chrome-stable para proveer /usr/bin/x-www-browser (x-www-browser) en modo automático
update-alternatives: utilizando /usr/bin/google-chrome-stable para proveer /usr/bin/gnome-www-browser (gnome-www-browser) en modo automático
update-alternatives: utilizando /usr/bin/google-chrome-stable para proveer /usr/bin/google-chrome (google-chrome) en modo automático
Procesando disparadores para gnome-menus (3.13.3-11ubuntu1.1) ...
Procesando disparadores para desktop-file-utils (0.23-1ubuntu3.18.04.2) ...
Procesando disparadores para mime-support (3.60ubuntu1) ...
Procesando disparadores para man-db (2.8.3-2ubuntu0.1) ...
christopher@christopher-VirtualBox:~$
```

1.2 Instalación de Microsoft Edge.

Para instalar Microsoft Edge nos dirigiremos a la página oficial de Microsoft Edge, <https://www.microsoftedgeinsider.com/> y descargaremos el archivo .deb de la versión Beta de Edge.

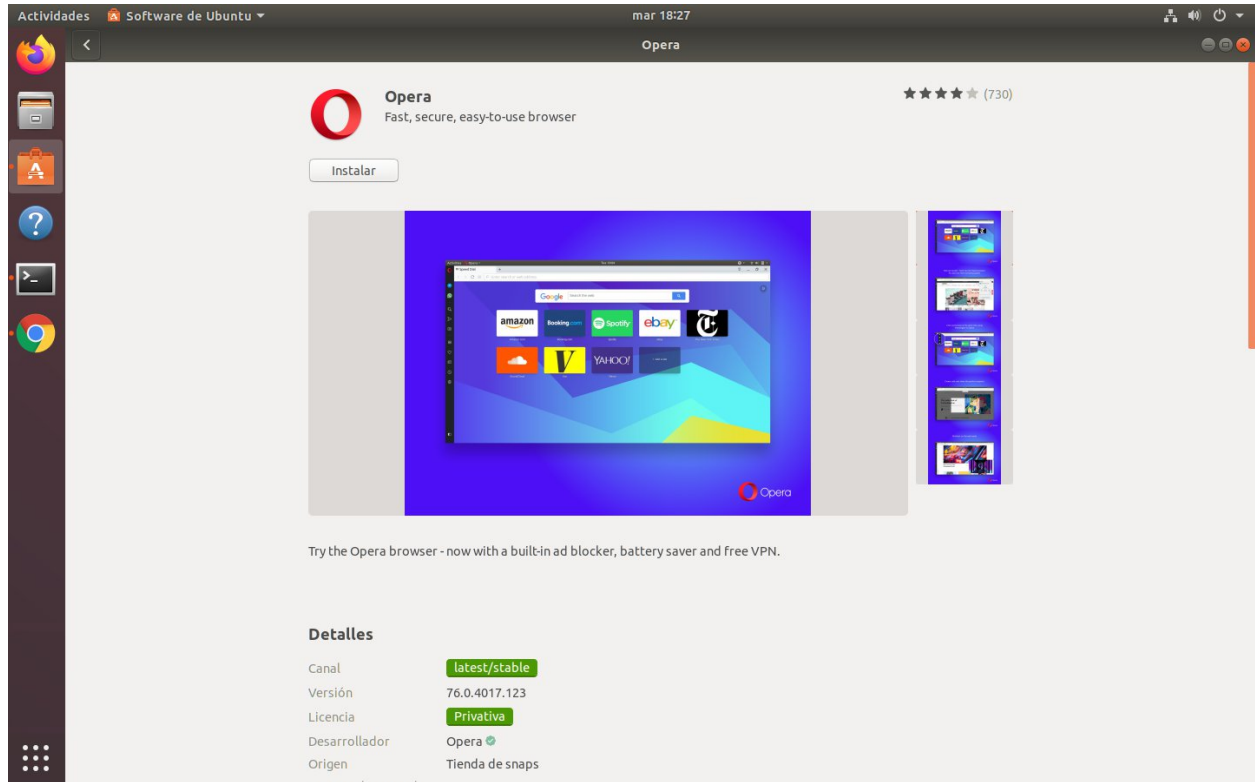


Una vez descargado haremos clic secundario en el archivo descargado y elegiremos la opción “Abrir con Instalación de Software”, esto instalará Edge en nuestro equipo.



1.3 Instalación de Opera.

Este navegador lo instalaremos con la aplicación de Software de Ubuntu. Sencillamente abriremos la aplicación, usaremos el buscador para encontrar Opera y la aplicación lo instalará automáticamente.



Como podemos comprobar, todos los navegadores están instalados en nuestro equipo.



2. Instalación del editor de páginas web

Como editor de páginas web vamos a instalar Visual Studio Codium. Esto es una variante de Visual Studio Code de Microsoft, donde usan el código fuente de Visual Studio con licencia MIT y le quitan los elementos de rastreo y telemetría que tiene la versión oficial de Microsoft.

Lo instalaremos usando la terminal de Ubuntu. Primero tenemos que añadir la clave GPG del repositorio donde se encuentra almacenado:

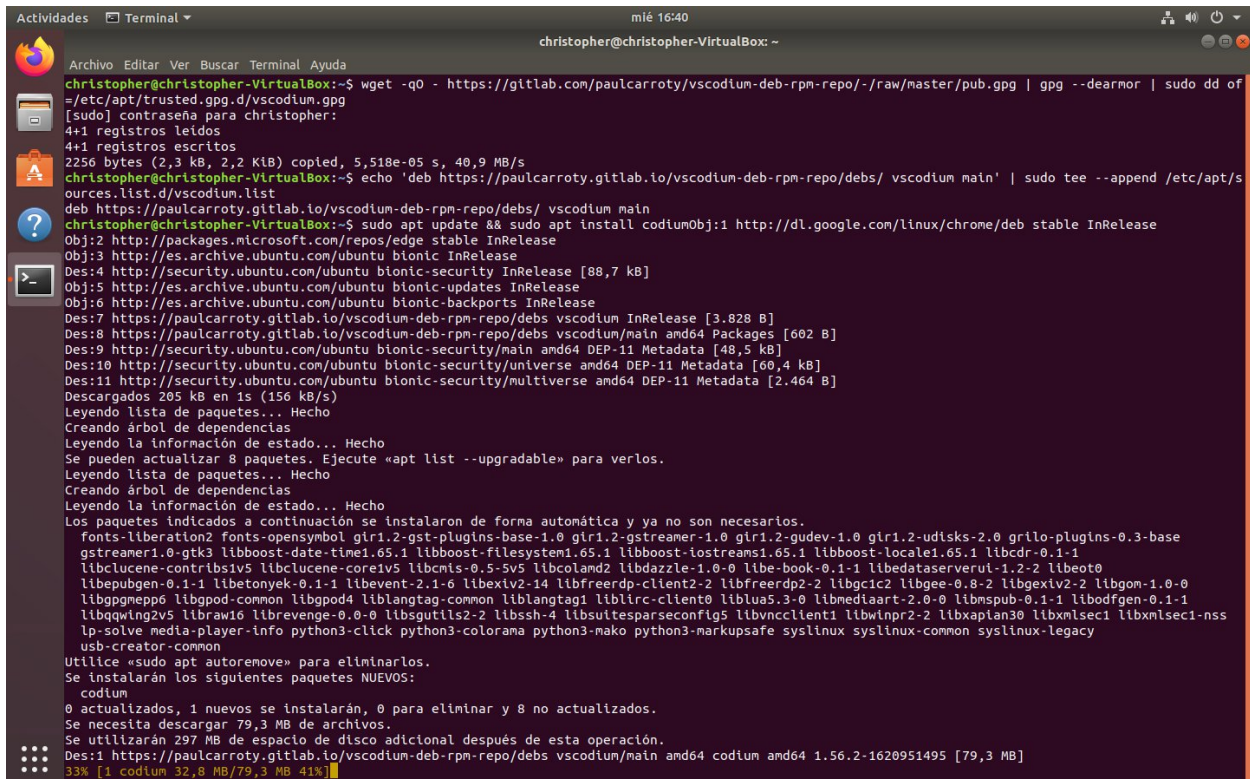
- o `wget -qO - https://gitlab.com/paulcarroty/vscodium-deb-rpm-repo/-/raw/master/pub.gpg | gpg --dearmor | sudo dd of=/etc/apt/trusted.gpg.d/vscodium.gpg`

Después añadiremos los repositorios con el siguiente comando:

- o `echo 'deb https://paulcarroty.gitlab.io/vscodium-deb-rpm-repo/debs/ vscodium main' | sudo tee --append /etc/apt/sources.list.d/vscodium.list`

Por último, actualizaremos la lista de repositorios e instalaremos el programa con el comando:

- o `sudo apt update && sudo apt install codium`



```
Actividades Terminal
mié 16:40
christopher@christopher-VirtualBox: ~
christopher@christopher-VirtualBox:~$ wget -qO - https://gitlab.com/paulcarroty/vscodium-deb-rpm-repo/-/raw/master/pub.gpg | gpg --dearmor | sudo dd of
-/etc/apt/trusted.gpg.d/vscodium.gpg
[sudo] contraseña para christopher:
4+1 registros leídos
4+1 registros escritos
2256 bytes (2,3 kB, 2,2 KiB) copied, 5,518e-05 s, 40,9 MB/s
christopher@christopher-VirtualBox:~$ echo 'deb https://paulcarroty.gitlab.io/vscodium-deb-rpm-repo/debs/ vscodium main' | sudo tee --append /etc/apt/s
ources.list.d/vscodium.list
deb https://paulcarroty.gitlab.io/vscodium-deb-rpm-repo/debs/ vscodium main
christopher@christopher-VirtualBox:~$ sudo apt update && sudo apt install codium
Obj:1 http://dl.google.com/linux/chrome/deb stable InRelease
Obj:2 http://packages.microsoft.com/repos/edge stable InRelease
Obj:3 http://es.archive.ubuntu.com/ubuntu bionic InRelease
Des:4 http://security.ubuntu.com/ubuntu bionic-security InRelease [88,7 kB]
Obj:5 http://es.archive.ubuntu.com/ubuntu bionic-updates InRelease
Obj:6 http://es.archive.ubuntu.com/ubuntu bionic-backports InRelease
Des:7 https://paulcarroty.gitlab.io/vscodium-deb-rpm-repo/debs vscodium InRelease [3.828 B]
Des:8 https://paulcarroty.gitlab.io/vscodium-deb-rpm-repo/debs vscodium/main amd64 Packages [602 B]
Des:9 http://security.ubuntu.com/ubuntu bionic-security/main amd64 DEP-11 Metadata [48,5 kB]
Des:10 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 DEP-11 Metadata [60,4 kB]
Des:11 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 DEP-11 Metadata [2.464 B]
Descargados 205 kB en 1s (156 kB/s)
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se pueden actualizar 8 paquetes. Ejecute «apt list --upgradable» para verlos.
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
fonts-liberation2 fonts-opensymbol gir1.2-gst-plugins-base-1.0 gir1.2-gstreamer-1.0 gir1.2-gudev-1.0 gir1.2-udisks-2.0 grilo-plugins-0.3-base
gstreamer1.0-gtk3 libboost-date-time1.65.1 libboost-filesystem1.65.1 libboost-iostreams1.65.1 libboost-locale1.65.1 libbcr-0.1-1
libclucene-contribs1v5 libclucene-core1v5 libcmis-0.5-5v5 libcolamd2 libdazzle-1.0-0 libe-book-0.1-1 libedataseerui-1.2-2 libeot0
libepubgen-0.1-1 libetonyek-0.1-1 libevent-2.1-6 libexiv2-14 libfreerdp-client2-2 libfreerdp2-2 libgic2 libgee-0.8-2 libgexiv2-2 libgom-1.0-0
libgpgmepp6 libgpod-common libgpod4 liblangtag-common liblangtag1 liblirc-client0 liblua5.3-0 libmediaart-2.0-0 libmshelp-0.1-1 libodfgen-0.1-1
libqimg2v5 libraw16 librevenge-0.0-0 libsgutil2-2 libssh-4 libsuitesparseconfig5 libvncclient1 libwinpr2-2 libxapian3 libxmlsec1 libxmlsec1-nss
lp-solve media-player-info python3-click python3-colorama python3-mako python3-markupsafe syslinux syslinux-common syslinux-legacy
usb-creator-common
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes NUEVOS:
codium
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 8 no actualizados.
Se necesita descargar 79,3 MB de archivos.
Se utilizarán 297 MB de espacio de disco adicional después de esta operación.
Des:1 https://paulcarroty.gitlab.io/vscodium-deb-rpm-repo/debs vscodium/main amd64 codium amd64 1.56.2-1620951495 [79,3 MB]
33% [1 codium 32,8 MB/79,3 MB 41%]
```


Cuando termine, VSCodium estará instalado en nuestro equipo.



La razón por la que usamos Visual Studio Codium es, aparte de por ser software libre de código abierto, por que de los IDEs actuales es el que ofrece mayor funcionalidad gracias a la enorme comunidad que desarrolla plugins.

Para el desarrollo de páginas y aplicaciones web hay tres plugin en concreto que nos facilitan mucho la vida:

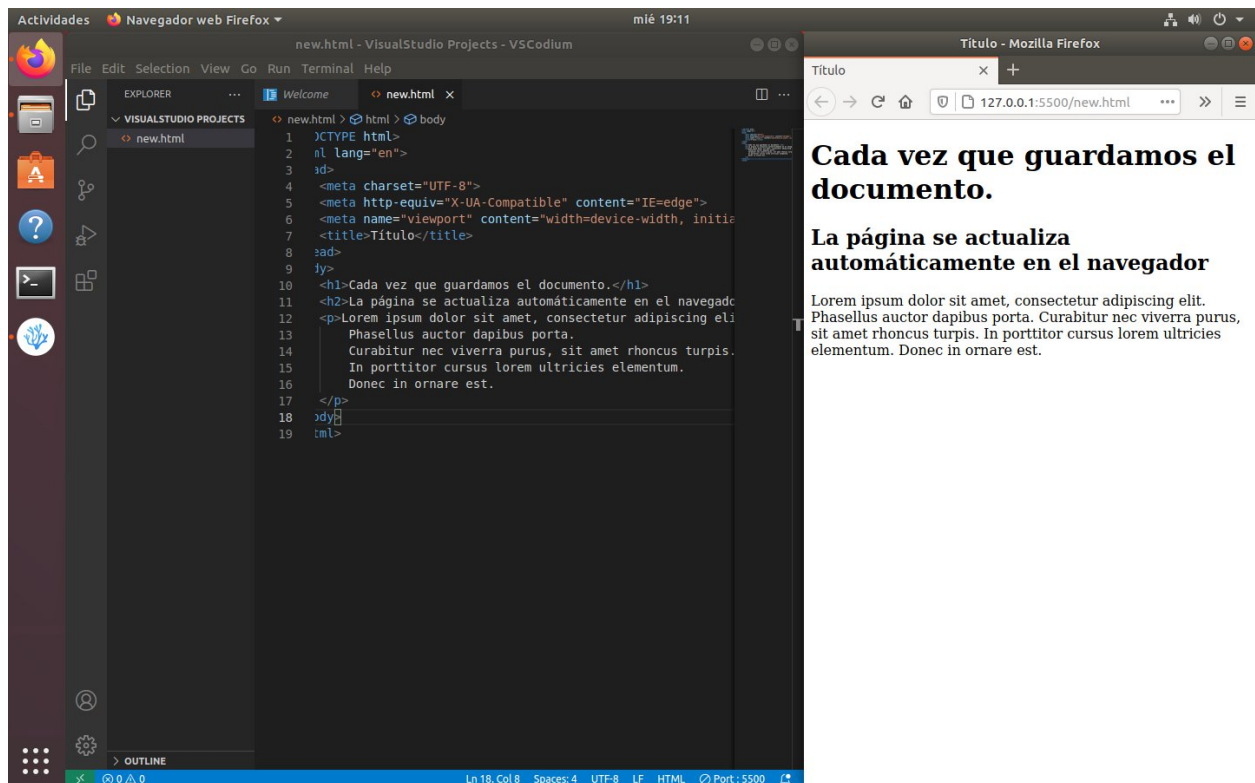
- **Bracket Pair Colorizer.** Este plugin dará colores distintos a nuestros paréntesis, así siempre sabremos qué abarca cada paréntesis en caso de que haya muchos metidos unos dentro de otros.

```
JS new.js > ...
1  var asda = 1;
2  var dasa = 2;
3
4  var sada = (asda+(dasa+2*((asda+dasa)*(asda-dasa))))
```

- **JavaScript ES6 code snippets.** Este plugin nos permite escribir estructuras comunes de JavaScript con comandos sencillos. Por ejemplo, si queremos crear un “getter”, solo tendremos que escribir las letras “pge” y pulsar el tabulador. Si queremos crear un “setter”, solo tendremos que escribir las letras “pse” y pulsar el tabulador.

```
JS new.js
1  set propertyName(value) {
2      ;
3  }
4
5  get propertyName() {
6      return this.;
7  }
```

- **Live Server.** Este plugin hace que Visual Studio actúe como servidor local, lo que nos permitirá abrir nuestro código directamente en el navegador y cada vez que pulsemos la tecla “guardar” nuestra página se actualizará automáticamente. Este plugin, sumado al uso de dos monitores o repartiendo el espacio de trabajo entre IDE y navegador, agilizará mucho el desarrollo de aplicaciones y páginas.



Otra funcionalidad de Visual Studio es la de escritura automática de código. Si vamos a empezar un proyecto nuevo en HTML, simplemente con escribir una exclamación cerrada ("!") y pulsar el tabulador se nos creará toda la estructura de HTML. Si queremos apuntar a un fichero CSS solo tendremos que escribir "link" y pulsar el tabulador. Los tags comunes de HTML se pueden escribir automáticamente siguiendo el mismo proceso de escribir unos pocos caracteres y pulsar el tabulador

En mi opinión la principal ventaja de Visual Studio sobre otros IDEs es que ha sabido adaptarse a los tiempos. Mientras que otros IDEs como NetBeans o Eclipse siguen siendo muy populares y siguen funcionando a la perfección, da la impresión de que ya no reciben el mismo cuidado por parte de los desarrolladores y la comunidad, lo cual nos deja con unos productos poco pulidos y que parece que fueron abandonados hace años. En cambio Visual Studio tiene un aspecto considerablemente más moderno (el tema oscuro de serie es muy de agradecer), una serie de funciones que agilizan mucho el trabajo y su sección de plugins no tiene rival.

3. Validación de www.google.com

Usando la dirección de validación del W3C indicada en los apuntes realiza la validación de la página de Google e indica los tipos de errores encontrados, cita 3 errores detectados y la solución propuesta a cada uno de ellos.

Los errores que encontramos al validar la página se dividen en dos categorías: **advertencias y errores**.

Las advertencias nos dan consejos para mejorar el rendimiento del código de la página, como eliminar información redundante o información que podemos añadir para que no haya lugar a dudas.

Los errores nos advierten de problemas de seguridad, falta de información importante dentro de nuestro código o de la presencia de código no permitido.

A continuación, veremos 3 de los 96 errores encontrados en la página.

Error 1:



El atributo "align" como tal no existe, tendría que ser sustituido por "align-content", "align-items", "align-self", "text-align" o "vertical-align", dependiendo de lo que se quiera.

Error 2:

11. **Warning** The `dialog` element is not supported in all browsers. Please be sure to test, and consider using a polyfill.
From line 142, column 26074 to line 142, column 26112
`x}</style><dialog class="spch-dlg" id="spch-dlg"><div c`

Aquí nos advierte de que el elemento “dialog” no está soportado por todos los navegadores (lo soportan todos menos Safari, que lo tiene deshabilitado de serie aunque se puede habilitar).

Nos recomienda usar “Polyfill”, es decir, código que se ejecuta dependiendo de si está soportado por el navegador o no.

Error 3:

29. **Error** Element `input` is missing required attribute `aria-expanded`.
From line 142, column 29028 to line 142, column 29365
`sw"></div><input class="glFyf gsfi" jsaction="paste:puy29d;" maxlength="2048" name="q" type="text" aria-autoco...se" title="Search" value="" aria-label="Search" data-ved="0ahUKewj11s563tPzAhXcTDABHFOLCG4Q39UDCAQ"></div>`
Attributes for element `input`:
Global attributes
`accept` when type is `file`
`alt` when type is `image`
`autocomplete` when type is `text`, `search`, `url`, `tel`, `email`, `password`, `date`, `month`, `week`, `time`, `datetime-local`, `number`, `range`, or `color`
`autofocus`

Aquí un error extraño. Nos dice que la barra de búsqueda de Google necesita el atributo “aria-expanded”, pero éste solo se utiliza para elementos desplegados, como menús. Este atributo no afecta visualmente o funcionalmente a la página, es para que los lectores de pantallas puedan saber si el elemento está desplegado o contraído, pero la barra de búsqueda de Google no se despliega. Supongo que se podría añadir el atributo y decir que está desplegado.