



Preparación del entorno de trabajo

Instalación del VSCode, Extensión LiveServer, NodeJS, Git y GitHub

Javier Ribal del Río

2025-10-11

Table of contents

1 IDE	2
1.1 ¿Qué es una IDE?	2
1.2 ¿Cuántas IDE existen?	2
1.3 ¿Qué IDE utilizaremos nosotros?	2
1.3.1 Instalación de VScode	2
1.3.2 Familiarizarse con VSCode	3
1.4 Extensiones	3
1.4.1 LiveServer	3
2 Node.JS y NPM	4
2.1 ¿Qué es node JS?	4
2.2 ¿Qué es NPM?	4
3 Git y GitHub	4
3.1 ¿Qué es Git?	4
3.2 ¿Qué es GitHub?	5
3.3 Git frente GitHub	5
3.4 Uso de Git y GitHub	5
3.5 Instalación de Git y Cuenta de GitHub	5
3.6 Diagrama de trabajo de Git	6
3.6.1 Git en VScode	6

Contenido

- Concepto de IDE
- Instalación del **Visual Studio Code** (VSCode)
- Configuración del VSCode, extensión LiveServer
- Instalación de NodeJS y `npm`
- Introducción a Git y GitHub



1 IDE

1.1 ¿Qué es una IDE?

Las Integrated development environment (IDE) son aplicaciones diseñadas para facilitar la programación a los programadores.

Aunque no hay una lista de características propias de las IDE, suelen tener características comunes, por ejemplo:

- *Debugger* integrado en la interfaz
- Gestor de archivos
- *Editor de código inteligente*: autocompletados

1.2 ¿Cuántas IDE existen?

- Android Studio
- RStudio
- Eclipse

1.3 ¿Qué IDE utilizaremos nosotros?

Principalmente trabajaremos con VSCode una IDE de propósito general que configuraremos para el desarrollo frontend

1.3.1 Instalación de VScode

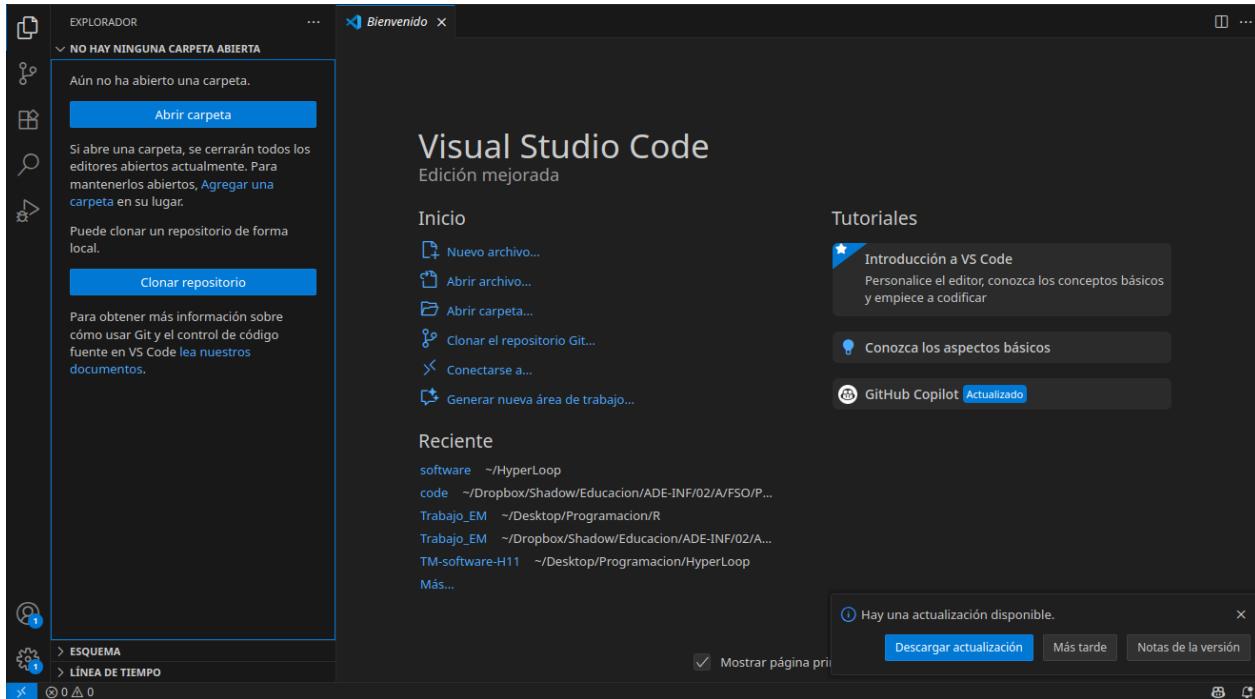
The screenshot shows the official Visual Studio Code website's download section. At the top, there's a navigation bar with links for Docs, Updates, Blog, API, Extensions, MCP, and FAQ, along with a search icon and a 'Download' button. Below the navigation, the title 'Download Visual Studio Code' is centered, with the subtitle 'Free and built on open source. Integrated Git, debugging and extensions.' underneath. Three large download buttons are displayed: one for Windows (Windows 10, 11), one for Linux (.deb for Debian, Ubuntu; .rpm for Red Hat, Fedora, SUSE), and one for Mac (macOS 11.0+). Each button has a small downward arrow icon. Below each button, there's a table of available installers or CLI options with their respective architectures (e.g., x64, Arm32, Arm64).

Platform	Architecture
Windows	x64, Arm32, Arm64
.deb	x64, Arm32, Arm64
.rpm	x64, Arm32, Arm64
.tar.gz	x64, Arm32, Arm64
Snap	Snap Store
CLI	x64, Arm32, Arm64

[Enlace](#)



1.3.2 Familiarizarse con VSCode



Barra lateral:

- Explorador de archivos
- Menú de Git
- Menú de extensiones
- Buscar
- *Debugger*

Editor o área de edición: componente más importante de la IDE, lugar donde se escribe el código.

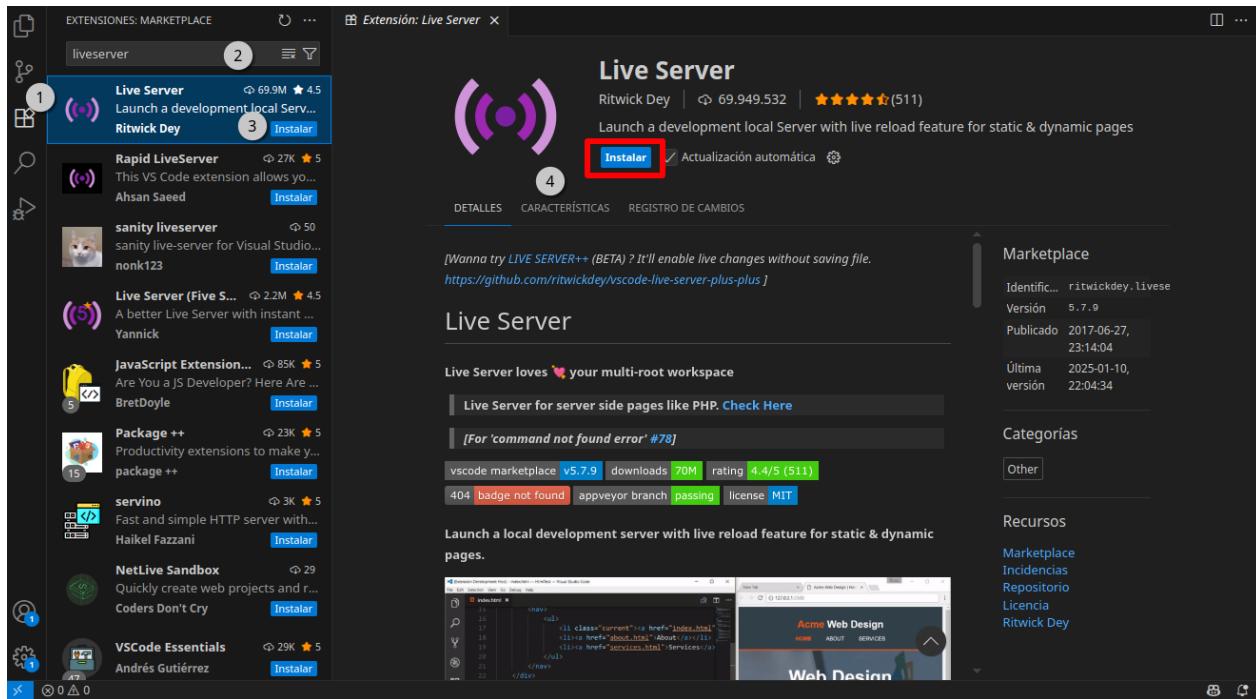
1.4 Extensiones

Las extensiones son aplicaciones software desarrolladas por empresas o por particulares para modificar o agregar una nueva funcionalidad a las IDE. Pueden ir desde pequeñas modificaciones del tema o de la UI, o incorporar nuevas funcionalidades como la interacción con bases de datos.

1.4.1 LiveServer

Nosotros para el transcurso del curso utilizaremos la extensión Liveserver que nos permitirá desplegar cómodamente un pequeño servidor para la programación con HTML (Ahondaremos más la próxima semana)

1.4.1.1 Instalación de LiveServer



1. Entrar en el menú de extensiones
2. Buscar por *liveserver*
3. Seleccionar extensión
4. Instalar
 - Al concluir la instalación debería de aparecer en la parte derecha de la barra de estado (barra interior)

2 Node.JS y NPM

2.1 ¿Qué es node JS?

Node.js es un **entorno de ejecución de JavaScript** basado en el motor V8 de Google Chrome. Permite ejecutar código JavaScript directamente en el sistema operativo, sin depender de un navegador. De este modo, el mismo lenguaje puede usarse tanto en el frontend como en el backend.

2.2 ¿Qué es NPM?

NPM (Node Package Manager) es el gestor de paquetes de Node.js. Facilita la instalación, actualización y gestión de librerías que amplían las funcionalidades de los proyectos.

Descarga node y npm nodejs.org

3 Git y GitHub

3.1 ¿Qué es Git?

Git es un sistema de control de versiones distribuido. Permite registrar los cambios de un proyecto, volver a versiones anteriores y trabajar de forma colaborativa sin sobrescribir el trabajo de otros. Cada usuario



dispone de una copia completa del repositorio, lo que facilita el trabajo sin conexión.

3.2 ¿Qué es GitHub?

GitHub es una plataforma en la nube que utiliza Git para almacenar y gestionar proyectos. Permite compartir código, colaborar con otros desarrolladores y mantener un control de versiones centralizado.

3.3 Git frente GitHub

- Git es la herramienta de control de versiones.(Software Libre, Linux Fundation)
- GitHub es la plataforma online donde se alojan los repositorios y se facilita el trabajo en equipo. (Corporativa Microsoft)

3.4 Uso de Git y GitHub

En Hyperloop UPV, tanto en los subsistemas de Software y Firmware utilizamos GitHub como herramienta de gestión de código.

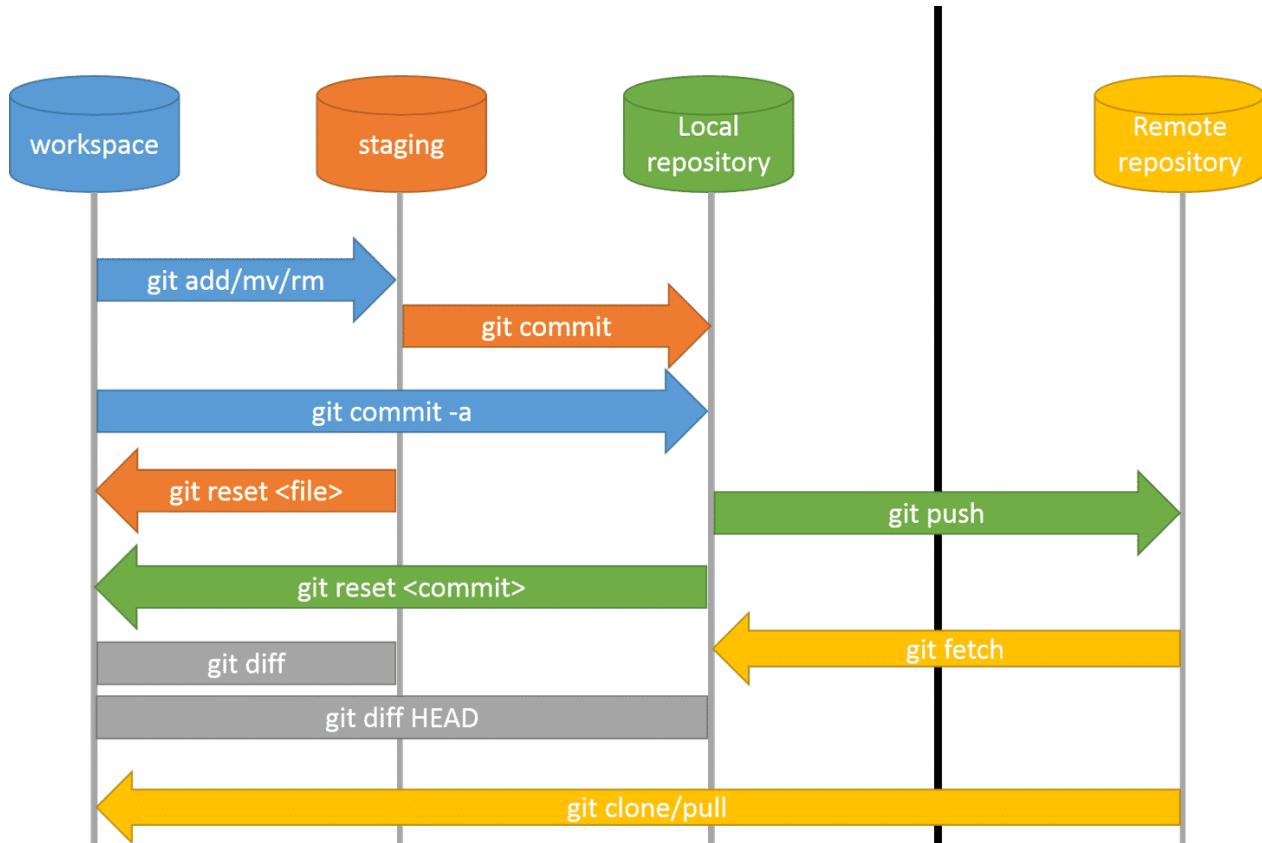
Nosotros en el TC, también trataremos de facilitarnos con esta herramienta

3.5 Instalación de Git y Cuenta de GitHub

- [Instalación de Git](#)
- [Cuenta de GitHub](#)



3.6 Diagrama de trabajo de Git



3.6.1 Git en VScode

