

# **Desarrollo Web - Cliente- Ud 1**

**Aprendizaje 1**

Para complementar tus estudios en programación web, estás valorando la idea de empezar a realizar algunos trabajos por tu cuenta, a modo de aprendizaje y portfolio.

Lo primero que tienes que hacer es tener claro las tecnologías y entornos que usarás, así que le pides ayuda a un compañero, y entre los dos realizáis un pequeño dossier con la comparativa de los diferentes entornos de desarrollo integrados (tanto gratuitos como de pago) disponibles.

Enumera sus características, licencias, ventajas e inconvenientes.

**Visual Studio Code (VS Code)**

**Características**

* Soporte para múltiples lenguajes (JavaScript, HTML, CSS, PHP, Python, etc.).
* Extensiones personalizables (Git, Docker, linters, debuggers).
* Integración con Git para control de versiones.
* Live Server para desarrollo web en tiempo real.

**Licencia**: Gratuita (Código abierto).

**Ventajas**

* Ligero y rápido.
* Gran cantidad de extensiones para personalizar el entorno.
* Comunidad activa y actualizaciones frecuentes.
* Funciona en Windows, macOS y Linux.

**Inconvenientes**

* Aunque es altamente personalizable, requiere la instalación de muchas extensiones para funcionalidades más avanzadas.

**Sublime Text**

**Características**

* Editor de texto rápido y ligero.
* Compatible con lenguajes web como HTML, CSS, JavaScript.
* Soporte de fragmentos de código (snippets) y múltiples cursores.
* Interfaz minimalista.

**Licencia**: Gratuita con limitaciones (licencia de pago opcional).

**Ventajas**

* Muy rápido, incluso con archivos grandes.
* Interfaz limpia y minimalista.
* Soporte para personalización mediante plugins.

**Inconvenientes**

* Funciones avanzadas limitadas en la versión gratuita.
* Menor cantidad de plugins que VS Code.

**JetBrains** **WebStorm**

**Características**

* IDE especializado en desarrollo web y frameworks como React, Angular, Vue.
* Integración avanzada con Git, depuración y testing.
* Autocompletado inteligente de código y herramientas de análisis.

**Licencia**: De pago (con prueba gratuita y descuentos para estudiantes).

**Ventajas**

* Herramientas integradas de gran calidad para depuración y testing.
* Perfecto para trabajar con JavaScript y frameworks modernos.
* Funciones inteligentes como el refactorizado de código y autocompletado avanzado.

**Inconvenientes**

* Requiere una licencia de pago (con descuentos).
* Mayor consumo de recursos que otros editores ligeros como Sublime Text o VS Code.

**Atom**

**Características**

* Editor de código con soporte para múltiples lenguajes y Git integrado.
* Extensible con paquetes y temas.
* Compatible con HTML, CSS, JavaScript, PHP, y más.

**Licencia**: Gratuita (Código abierto).

**Ventajas**

* Alta personalización y amplia gama de plugins.
* Interfaz amigable.
* Soporte para colaboración en tiempo real (Teletype).

**Inconvenientes**

* Consumo de recursos mayor que otros editores como Sublime Text.
* Menos actualizaciones desde que GitHub decidió centrar su desarrollo en otras herramientas.

**Brackets**

**Características**

* Editor enfocado en el desarrollo web frontend.
* Vista previa en vivo (Live Preview) para HTML y CSS.
* Integración con preprocesadores como Sass y Less.

**Licencia**: Gratuita (Código abierto).

**Ventajas**

* Ideal para desarrolladores frontend, con funciones integradas como previsualización en vivo.
* Interfaz limpia y fácil de usar.

**Inconvenientes**

* Menos extensible y flexible en comparación con otros editores como VS Code o Atom.
* Desarrollo del proyecto discontinuado (poca o nula actualización).

**NetBeans**

**Características**

* IDE de código abierto compatible con múltiples lenguajes (Java, PHP, HTML5, JavaScript).
* Buenas herramientas de depuración y testing integradas.
* Soporte para desarrollo web y aplicaciones empresariales.

**Licencia**: Gratuita (Código abierto).

**Ventajas**

* Robusto para proyectos grandes.
* Buen soporte para programación en múltiples lenguajes, especialmente Java.

**Inconvenientes**

* Pesado en comparación con editores ligeros como VS Code o Sublime.
* Interfaz menos moderna.

**Elección final: Visual Studio Code (VS Code)**

Después de revisar las características, ventajas e inconvenientes de los diferentes IDEs, he decidido optar por Visual Studio Code (VS Code) por los siguientes motivos:

**Extensibilidad y personalización**: VS Code ofrece una gran cantidad de extensiones para personalizar el entorno de desarrollo, lo que lo hace adecuado tanto para proyectos pequeños como grandes.

**Compatibilidad multiplataforma**: Funciona en Windows, macOS y Linux, lo que facilita la colaboración entre desarrolladores que puedan estar trabajando en diferentes sistemas operativos.

**Comunidad activa**: La comunidad de VS Code es muy activa, lo que asegura actualizaciones frecuentes y la disponibilidad de recursos de aprendizaje y soporte.

**Costo**: Es gratuito y de código abierto, lo que lo convierte en una excelente opción para quienes están comenzando y no quieren comprometerse con una licencia de pago.

**Integración con Git**: Las funciones de integración con Git son fáciles de usar y muy útiles para gestionar el control de versiones en proyectos web.