



Identificación: Anexo01-Entorno-Trabajo

Versión: 1.0

Tabla de contenido

<u>¿QUÉ ES UN ENTORNO DE DESARROLLO?</u>	<u>2</u>
NIVELES DEL ENTORNO DE DESARROLLO	2
SERVIDOR WEB	3
GIT	4
GITHUB	5
GITLAB	6
<u>EDITORES DE CÓDIGO Y/O IDE SUGERIDOS</u>	<u>7</u>
<u>MANUALES EXTERNOS:</u>	<u>7</u>
<u>BIBLIOGRAFIA.....</u>	<u>7</u>



¿QUÉ ES UN ENTORNO DE DESARROLLO?

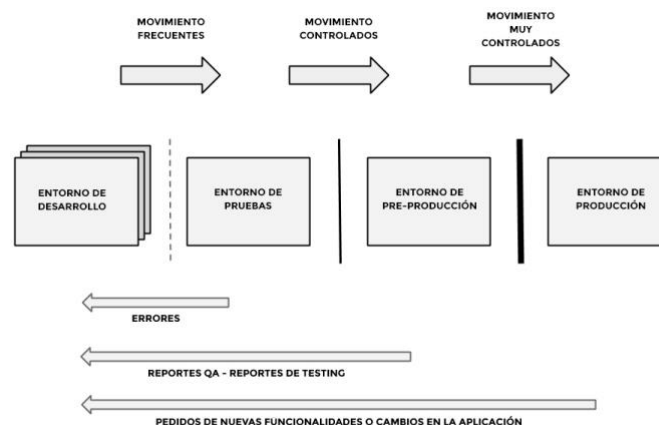
Un entorno de desarrollo es un conjunto de procedimientos y herramientas que se utilizan para desarrollar un código fuente o programa. Este término se utiliza a veces como sinónimo de entorno de desarrollo integrado (IDE), que es la herramienta de desarrollo de software utilizado para escribir, generar, probar y depurar un programa. También proporcionan a los desarrolladores una interfaz de usuario común (UI) para desarrollar y depurar en diferentes modos.

A la hora de elegir un entorno de desarrollo o IDE (Integrated Development Environment) es fundamental tener definido qué lenguaje de programación se va a utilizar tanto en el Frontend (la parte visible de la web) como en el Backend.

Niveles del entorno de desarrollo

El entorno de desarrollo normalmente tiene tres niveles de servidores, clasificados como desarrollo, montaje y producción. Los tres niveles juntos se denominan generalmente como el DSP.

- **Servidor de desarrollo:** Aquí es donde el desarrollador prueba el código y comprueba si la aplicación se ejecuta correctamente con ese código. Una vez que la implementación ha sido probada y el desarrollador considera que el código trabaja de forma correcta, la aplicación se mueve entonces al servidor intermedio de montaje.
- **Servidor de integración:** Este entorno se hace para que se vea exactamente igual que el entorno del servidor de producción. La aplicación se prueba en el servidor de ensayo para comprobar la fiabilidad y para asegurarse de que no falla en el servidor de producción real. Este tipo de pruebas en el servidor intermedio es el último paso antes de que la aplicación se despliegue en un servidor de producción. La aplicación tiene que ser aprobada con el fin de implementarla en el servidor de producción.
- **Servidor de producción:** Una vez realizada la aprobación, la aplicación se convierte en una parte de este servidor.





REQUERIMIENTOS

Es necesario tener instalado el servidor para ejecutar aplicaciones, interprete del lenguaje, motor de bases de datos y gestor de dependencias.

Servidor Web

Instalar Cualquier de entornos de desarrollo con PHP

Multiplataforma



<https://www.apachefriends.org/es/index.html>



<https://www.wampserver.com/en/>

Windows



<https://laragon.org/>

MAC OSX



Nota: La versión minima de php PHP 7.3.11 (cli) (built: Jun 5 2020 23:50:40) (NTS)



Motores de bases de datos, personalmente prefiero mysql, mariaDb o mongodb



Composer como gestor de dependencia (es fundamental)



Dependency Manager for PHP

<https://getcomposer.org/>

GIT



<https://git-scm.com/>

Git, que presenta una arquitectura distribuida, es un ejemplo de DVCS (sistema de control de versiones distribuido, por sus siglas en inglés). En lugar de tener un único espacio para todo el historial de versiones del software, como sucede de manera habitual en los sistemas de control de versiones antaño populares, como CVS o Subversion (también conocido como SVN), en Git, la copia de trabajo del código de cada desarrollador es también un repositorio que puede albergar el historial completo de todos los cambios.

- Es muy potente
- Fue diseñada por Linus Torvalds
- No depende de un repositorio central
- Es software libre
- Con ella podemos mantener un historial completo de versiones
- Podemos movernos, como si tuviéramos un puntero en el tiempo, por todas las revisiones de código y desplazarnos una manera muy ágil.
- Es muy rápida
- Tiene un sistema de trabajo con ramas que lo hace especialmente potente



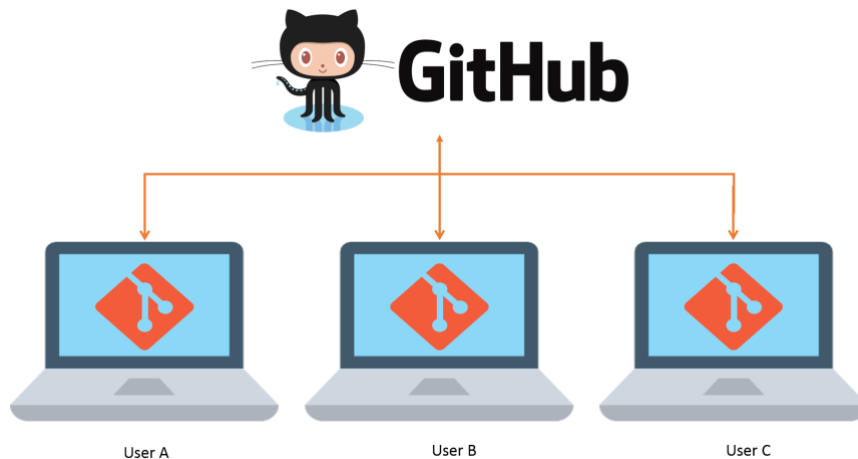
- En cuanto a la funcionalidad de las ramas, las mismas están destinadas a provocar proyectos divergentes de un proyecto principal, para hacer experimentos o para probar nuevas funcionalidades.
- Las ramas pueden tener una línea de progreso diferente de la rama principal donde está el core de nuestro desarrollo. En algún momento podemos llegar a probar algunas de esas mejoras o cambios en el código y hacer una fusión a nuestro proyecto principal, ya que todo esto lo maneja Git de una forma muy eficiente



GITHUB

En un nivel más alto, GitHub es un sitio web y un servicio en la nube que ayuda a los desarrolladores a almacenar y administrar su código, al igual que llevar un registro y control de cualquier cambio sobre este código. Para entender exactamente qué es GitHub, primero usted necesita conocer los dos principios que lo conectan:

- Control de versión
- Git



<https://github.com/>



GITLAB



<https://about.gitlab.com/>

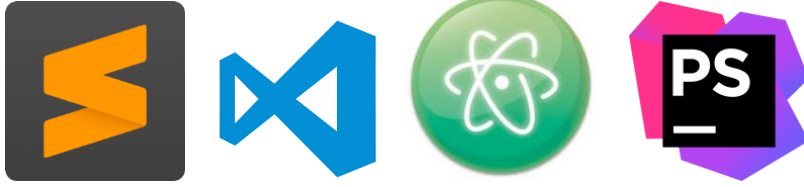
Gitlab es un servicio web de control de versiones y desarrollo de software colaborativo basado en Git. Además de gestor de repositorios, el servicio ofrece también alojamiento de wikis y un sistema de seguimiento de errores, todo ello publicado bajo una Licencia de código abierto.

GitLab es una suite completa que permite gestionar, administrar, crear y conectar los repositorios con diferentes aplicaciones y hacer todo tipo de integraciones con ellas, ofreciendo un ambiente y una plataforma en cual se puede realizar las varias etapas de su SDLC/ADLC y DevOps.

Fue escrito por los programadores ucranianos Dmitriy Zaporozhets y Valery Sizov en el lenguaje de programación Ruby1 con algunas partes reescritas posteriormente en Go, inicialmente como una solución de gestión de código fuente para colaborar con su equipo en el desarrollo de software. Luego evolucionó a una solución integrada que cubre el ciclo de vida del desarrollo de software, y luego a todo el ciclo de vida de DevOps. La arquitectura tecnológica actual incluye Go, Ruby on Rails y Vue.js.



EDITORES DE CÓDIGO Y/O IDE SUGERIDOS



Manuales Externos:

<https://www.mclibre.org/consultar/informatica/lecciones/vsc-instalacion.html>

Bibliografía

Instalación. Visual Studio Code. Informática. Bartolomé Sintés Marco. www.mclibre.org. (2021). Retrieved 1 February 2021, from <https://www.mclibre.org/consultar/informatica/lecciones/vsc-instalacion.html>

¿Conoces los entornos de trabajo utilizados en el desarrollo de una aplicación web?. (2021). Retrieved 1 February 2021, from <https://dharmawebstudio.com/en/blog/do-you-know-the-work-environments-used-in-the-development-of-a-web-application>