

DESARROLLO WEB PROFESIONAL UTILIZANDO EL ADMINISTRADOR DE VERSIONES GIT Y GITHUB.¹

Dr. Ángel Salvador López Vásquez²
angel.vasquez@zicatela.umar.mx

Abril 2017

En los últimos años mucho se comenta sobre el desarrollo web, pero muchas de estas veces hacemos a un lado la parte medular que es desarrollar a nivel profesional, por eso en este documento se brinda un punto importante en el seguimiento al control de versiones que debe llevar cada uno de los archivos involucrados en un proyecto.

¿Cuántos de ustedes están acostumbrados a desarrollar proyectos sin controlar o administrar las versiones de los mismos?, ¿Cuántos dolores de cabeza te ha ocasionado perder la última versión de tus archivos por motivos de no guardar a tiempo o sobrescribir los mismos con información errónea?, si eres de los programadores que guardan copias, copias y más copias de un archivo (.jsp, .php, .html, .py, .css, .js, .sql, etc.), o de tus directorios, entonces quiero decirte que no trabajas como un profesional, solo eres un aficionado más en el mundo del desarrollo. Pero no te preocupes eso puede cambiar, por esa razón comienzo con la redacción de este pequeño artículo y te comparto un poquito de la experiencia que tengo en el mundo del desarrollo utilizando GIT y GITHUB para controlar las versiones de mis proyectos, los cuales me han sido muy útiles tanto para fines académicos como profesionales.

Pero a todo esto, ¿qué es Git?, Git en concreto es un sistema de control de versiones distribuido que permite controlar tus archivos o conjunto de archivos a lo largo del tiempo, con la finalidad de que evites crear copias que generen redundancia o peor aún, sobrescribir archivos que te lleven a la pérdida de los mismos. ¿Y quién desarrolló git?, para darse una idea de lo valioso que es Git, solo basta con saber quién está detrás, dado que fue desarrollado

¹ El presente resumen es un producto final del seminario brindado en la Universidad del Mar Campus Puerto Escondido con fecha 26 de abril de 2017.

² Profesor-Investigador adscrito a la Lic. En Informática, de la Universidad del Mar Campus Puerto Escondido

Ángel Vásquez

ni más ni menos que por el Ingeniero de Software Finlandés Linus Benedict Torvalds conocido por iniciar y mantener el desarrollo del núcleo ‘kernel’ de Linux. ¿Y cuáles son sus características?, entre las características de Git se encuentra la rapidez en el desarrollo no lineal gestionado por ramas, gestión eficiente de proyectos grandes, dada la velocidad de gestión de diferencias entre archivos, entre otras mejoras de optimización.

Por otro lado, ¿qué es Github?, es una plataforma que nos va a permitir alojar nuestros proyectos utilizando el control de versiones Git, además de brindarte herramientas muy útiles para el trabajo en equipo dentro de tu proyecto, Github utiliza el framework de ruby on rails³. Prácticamente todo el control de versiones de nuestros archivos con Git, podrán ser enviados mediante SSH a Github, esto tiene muchas ventajas, de igual manera si no quieres que tus repositorios sean consultados por alguien más, podrás migrar a una cuenta de pago de github, así proteges tus repositorios y solo tu o tu equipo de trabajo tendrá acceso a el.

Usando Git y Github puedes contribuir a mejorar el software de alguien más o en su caso trabajar en conjunto con otros desarrolladores, para alcanzar esta meta Github posee funcionalidades para hacer un ‘fork’⁴ y solicitar ‘pulls’⁵ de los repositorios. Realizar un fork es simplemente clonar un repositorio ajeno a tu cuenta con la finalidad de mejorar o eliminar algún bug que encuentres, una vez realizadas las modificaciones puedes enviar un ‘pull’ al dueño del proyecto, éste podrá analizar los cambios que realizaste, y si considera interesante tu contribución lo podrá adjuntar al repositorio principal.

Ángel Vásquez

³ Ruby on Rails, también conocido como RoR o Rails, es un framework de aplicaciones web de código abierto escrito en el lenguaje de programación Ruby, siguiendo el paradigma del patrón Modelo Vista Controlador.

⁴ Fork - Hacerle un fork a un repositorio significa copiarlo, con la finalidad de proponer cambios al proyecto original.

⁵ Pull – Comando utilizado en git para fusionar otras ramas en la rama activa, por ejemplo: Te encuentras en tu rama principal máster y necesitas actualizar los últimos cambios remotos hacia esta, tendríamos que escribir: git pull origin nombreRama.

Git al ser un sistema de control de versiones distribuidas está preparado para trabajar en equipos ‘distribuidos’ es decir, podrás trabajar desde el patio de tu casa, oficina, aeropuertos, cafés, etc., de forma eficiente. Ahora, imagínate que te encuentras en México y tu equipo de trabajo en Francia, ¿cómo le harán para coordinar el desarrollo de un proyecto grande que se tiene con alguno de sus clientes potenciales?, o ¿cómo saber que parte del código han tocado y viceversa?. Estos problemas Git los resuelve sin mayor problema, con herramientas tal vez un poco complicadas de configurar si no tienes la experiencia y conocimientos adecuados (servidores ssh, claves públicas, claves privadas, bash, etc.), pero una vez que te encaminas a usar Git, creeme nada te detiene, bueno salvo que cometas algún error, por eso es importante que comiences a usar un buen controlador de versiones, por el lado contrario Github te facilita toda la infraestructura para trabajar en equipos distribuidos a través de una interfaz web gráfica de lo más cómoda, o en su caso utilizando la aplicación para escritorio disponible únicamente para Windows y Mac. Pero desde mi punto de vista te recomiendo iniciar con Git que sería como aprender a manejar un vehículo estándar y una vez que te encamines a ser un buen piloto podrás alternar y usar Github como si fuera tu vehículo automático.

¿Entonces te animas a usar un control de versiones como Git y Github para el desarrollo de proyectos?.

Ángel Vásquez