

Universidad de San Carlos de Guatemala

Facultad de Ingeniería

Escuela de Ciencias



Manual de Uso de GitHub

Hilbert Josué Perucho Ramírez

201313889

## ¿Qué es Git?

Git es un software de **control de versiones**. El control de versiones, resumiéndolo mucho, es la gestión de los diversos cambios que se realizan sobre un repositorio (un repositorio es el nombre que recibe el lugar donde se aloja el código de un proyecto de desarrollo en algún lenguaje de programación).

## Objetos de Git

Hay 4 tipos de objetos en git (el más importante a entender es el commit).

1. Blob: se usa para almacenar datos de archivos, es generalmente un archivo.
2. Tree: es, básicamente, como un directorio, hace referencia a un conjunto de otros *trees* y/o *blobs*(por ej. archivos y subdirectorios).
3. Commit: apunta a un determinado *tree*, marcando como era en un momento determinado (quien no haya entendido lo que es un *tree*, sustituya la palabra *tree* por *archivo*). Contiene información sobre ese momento determinado, los cambios del autor desde el último commit, el commit anterior (conocido como *parent*), etc. También se puede entender un commit, de una forma más imprecisa y coloquial, como la modificación o el conjunto de modificaciones a uno o varios archivos del repositorio. Otra forma de entenderlo también sería, como una "foto" de uno o varios archivos del repositorio en un momento determinado.
4. Tag: es una forma de marcar un commit como específico de alguna forma. Se usa normalmente para marcar algunos commits como releases específicos o algo destacable en esas líneas.

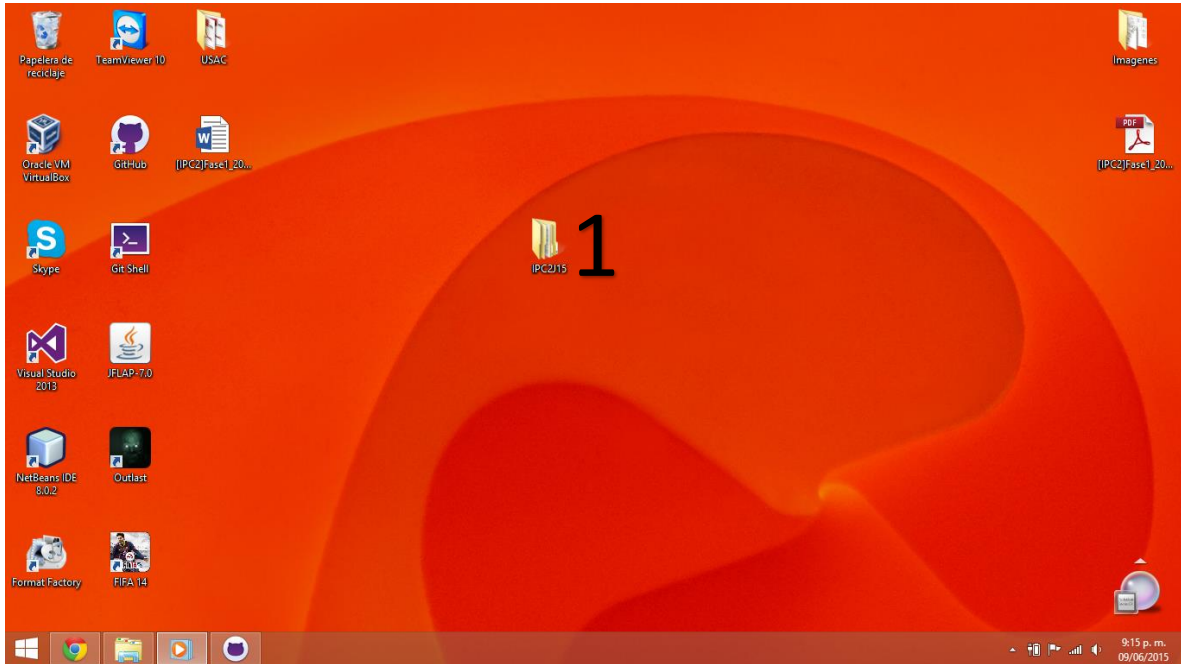
## ¿Qué es GitHub?

Github es una plataforma de desarrollo colaborativo de software para **alojar proyectos** usando el sistema de control de versiones Git. El código se almacena de forma pública, aunque también se puede hacer de forma privada, creando una cuenta de pago. También se pueden obtener repositorios privados (de pago) si se es estudiante.

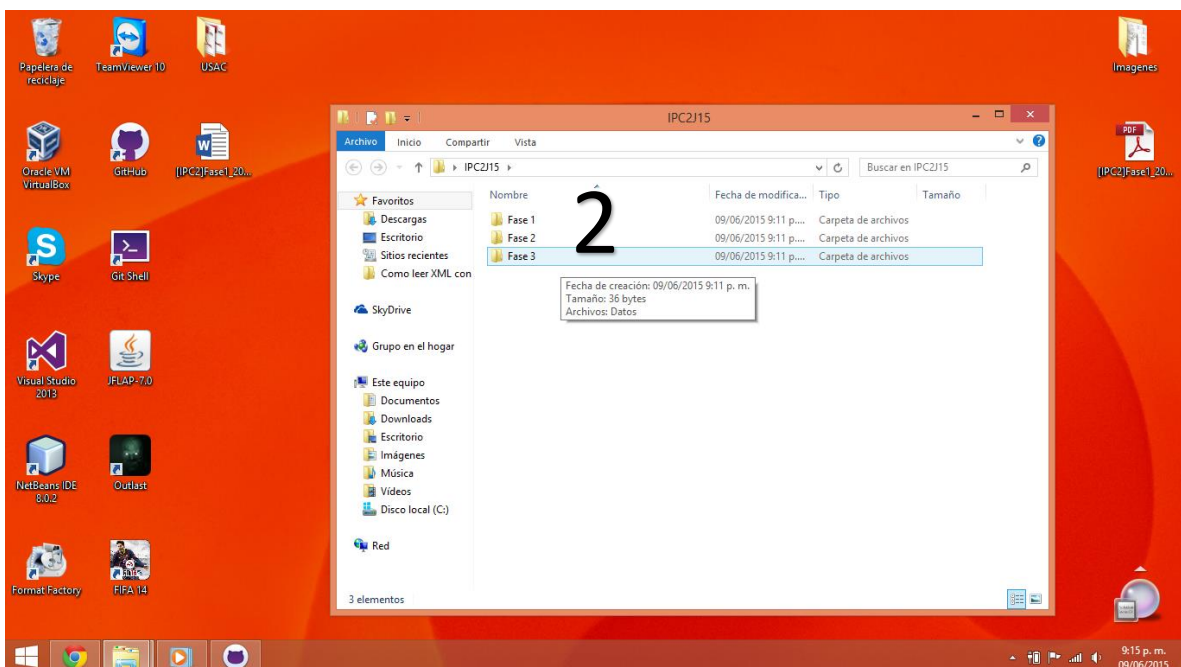
Github no sólo ofrece alojamiento del código si no muchas más posibilidades asociadas a los repos como son, forks, issues, pull requests, diffs, etc. Se verán todos con detalle más adelante.

# Manual de uso de GitHub

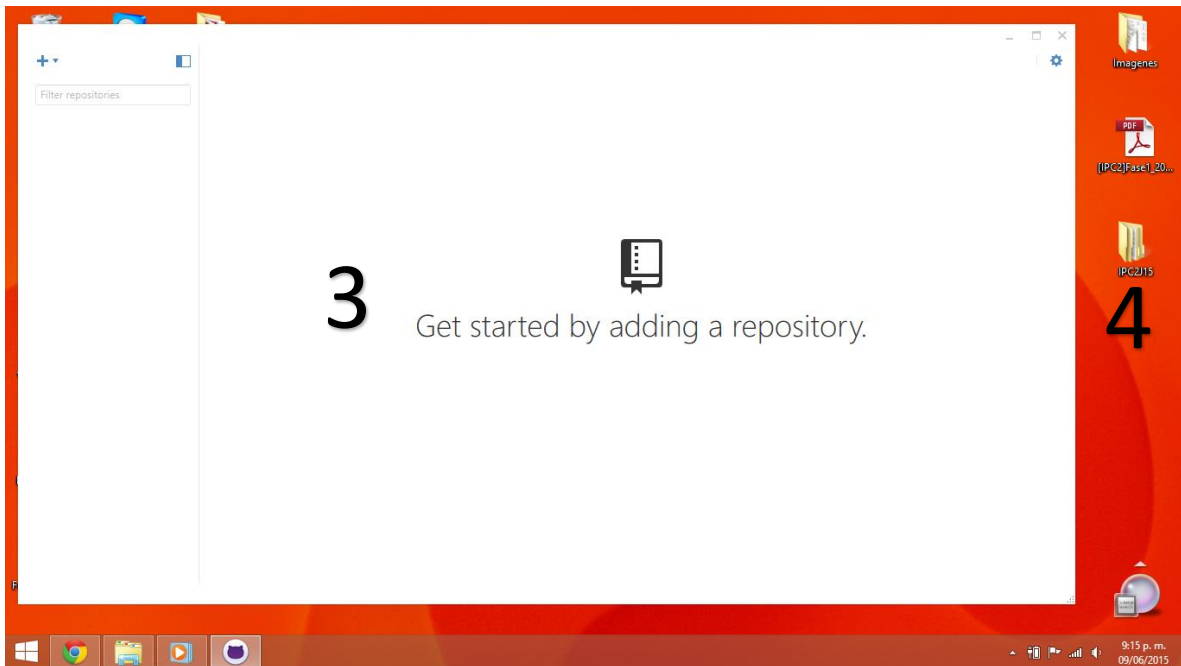
1. Creación de carpeta en el escritorio en cualquier otro nombre, este será el repositorio, pero antes asegúrese de descargar la aplicación de GitHub para Windows o para el sistema operativo con el que cuente.



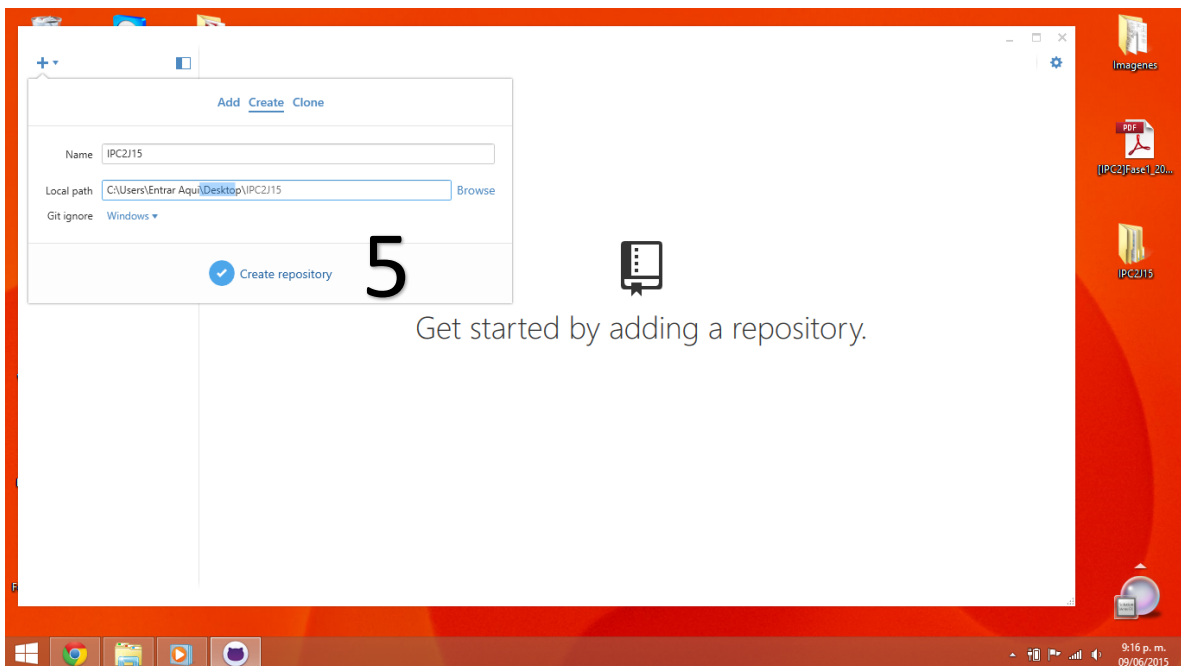
2. Segundo se creara las carpetas que irán dentro del repositorio y la vez agregarle un archivo .txt con cualquier información para poder realizar este paso ya que con eso me funciono a mí.



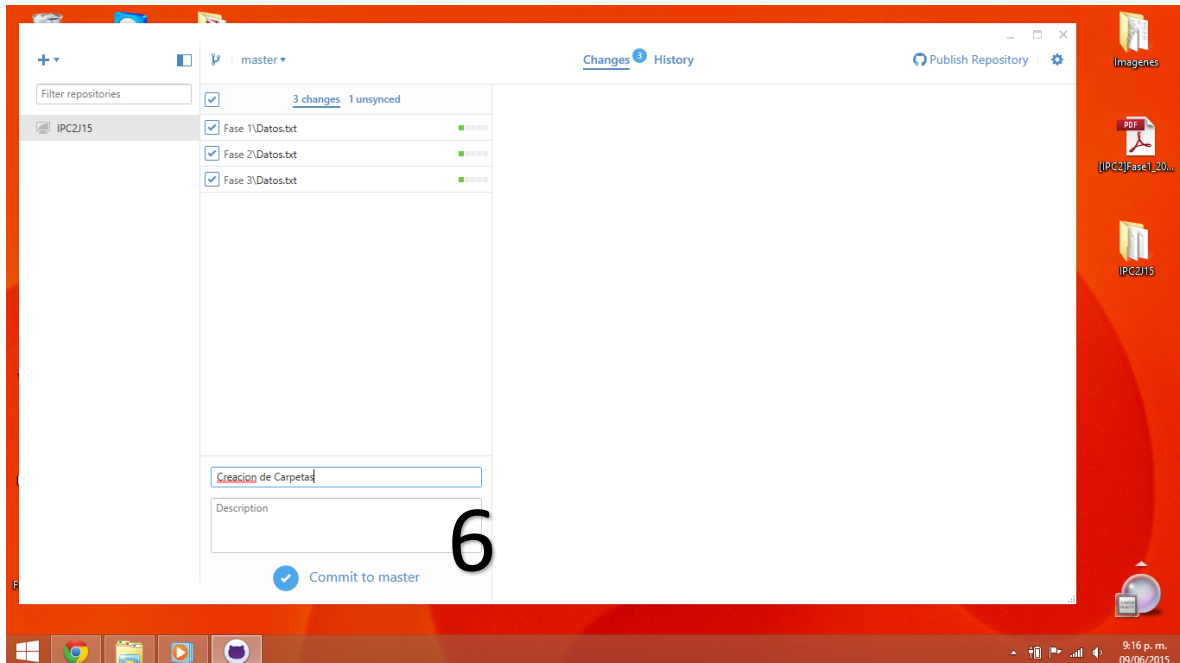
3. Abrir la aplicación GitHub y cuando se abra por primera vez pedirá iniciar sesión entonces hacer ese paso para que no haya ningún problema.
4. Arrastrar la carpeta Creada con anterioridad hacia la aplicación GitHub



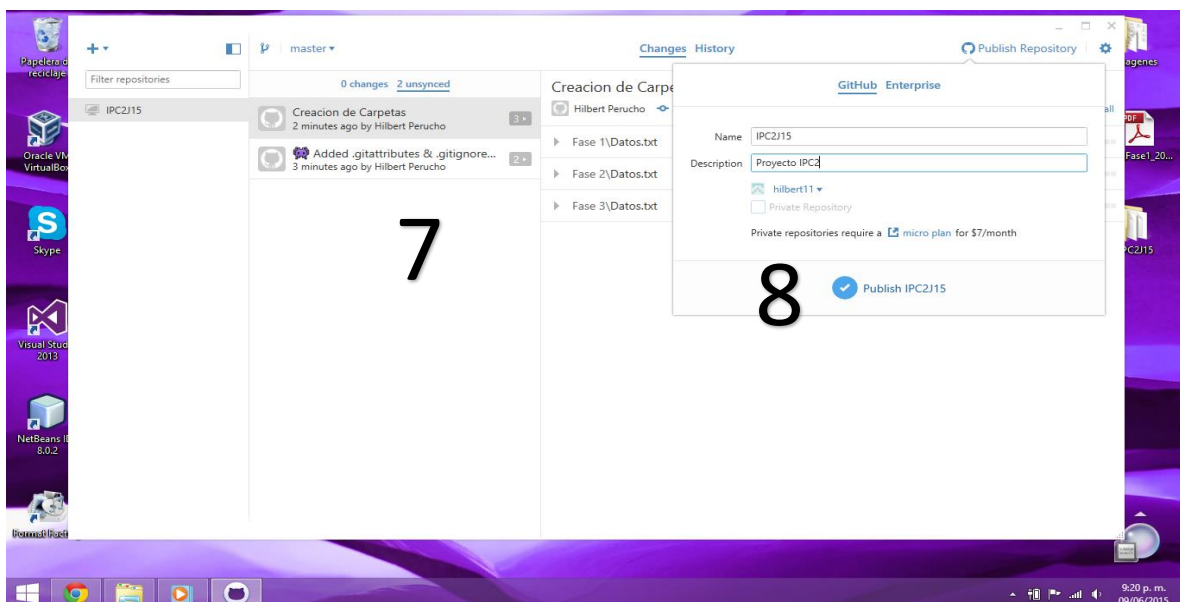
5. Cuando se arrastre la carpeta solo tomara referencia de la ubicación de la carpeta y desplegara lo siguiente que será el crear el Repositorio. Se Coloca el nombre que normalmente agarra el nombre de la carpeta con el que se creó con anterioridad. Cuando ya se tenga todo entonces se le da "Create Repository"



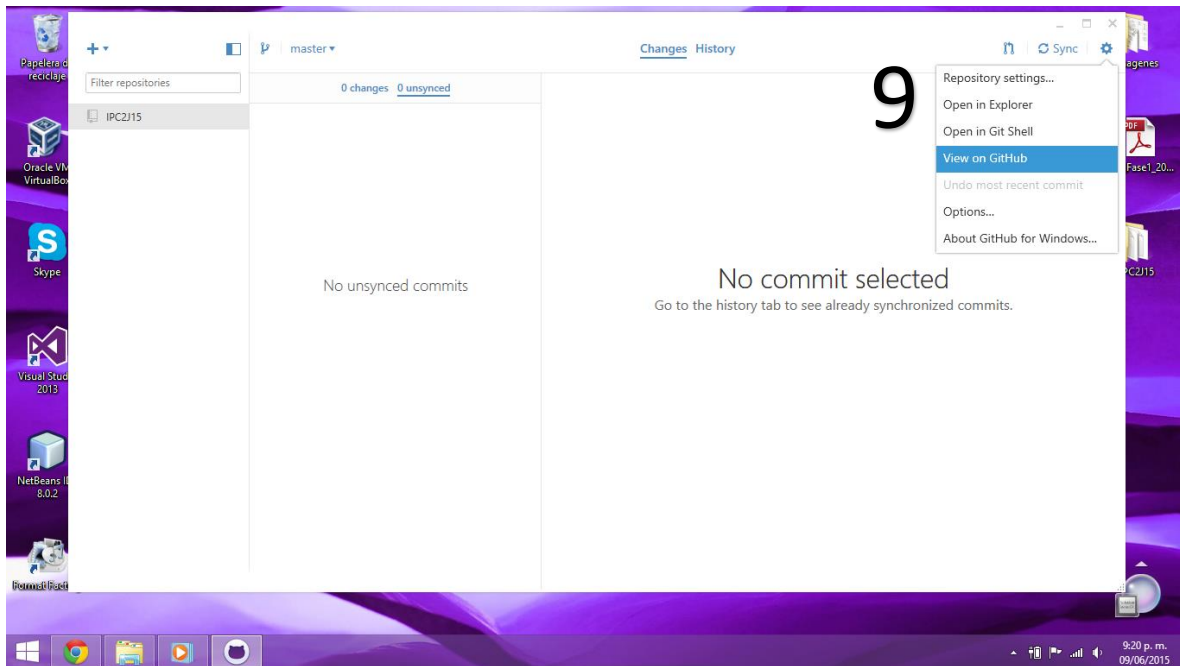
6. La aplicación tomara por si sola las cosas que contengan las carpetas y se mostrar cómo se puede apreciar. Ya con esto En la parte se debe de hacer un Commit Esto se refiere como a un tipo de nombre de guardado con el que todos tendrá acceso a la hora de que el repositorio sea público que normalmente Se pone así. Presionar “Commit to master”.



7. En esta parte mostrara las carpetas creadas que son los repositorios. De esta misma manera se agregaran carpetas o algún tipo de archivo que se desee.
8. Ya teniendo todo lo que se cargara al principio que es únicamente el Repositorio Principal Habrá que darle Click en la parte donde dice “Publish Repository” cuando se de click en esa parte se procederá a agregarle una descripción y a darle click en “Publish (Nombre de repositorio)”



9. En esta parte se dará click en la tuerca que aparece de lado derecho en la parte superior y luego se procede a dar click en “View on GitHub” esto para verificar que los archivos si fueron cargados correctamente.



10. Ya por ultimo aparecerá de esta manera en el explorador que tenga predeterminado y con esto se concluye de que si fue creado con éxito el Repositorio en GitHub

