a. Control de versiones (VC)

 El control de versiones es un sistema que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que puedas recuperar versiones específicas más adelante. A pesar de que los ejemplos de este libro muestran código fuente como archivos bajo control de versiones, en realidad cualquier tipo de archivo que encuentres en un ordenador puede ponerse bajo control de versiones.

b. Control de versiones distribuido (DVC)

los clientes no sólo descargan la última instantánea de los archivos: replican completamente el repositorio. Así, si un servidor muere, y estos sistemas estaban colaborando a través de él, cualquiera de los repositorios de los clientes puede copiarse en el servidor para restaurarlo. Cada vez que se descarga una instantánea, en realidad se hace una copia de seguridad completa de todos los datos

c. Repositorio remoto y Repositorio local

d. Copia de trabajo / Working Copy

Dentro de git es possible clonar un commit con el commando git clone

e. Área de Preparación / Staging Area

Básicamente se trata de la misma información dada por el comando 'git status', pero mas sucinta e informativa. Se ve una lista de cambios ya preparados, en la izquierda; y de los que están aún sin preparar, en la derecha.

f. Preparar Cambios / Stage Changes

para preparar archivos, y para otras cosas como marcar como resueltos archivos con conflictos de unión—). Ejecutamos git add para preparar el archivo benchmarks.rb, y volvemos a ejecutar git status:

g. Confirmar cambios / Commit Changes

Al confirmer los cambios

h. Commit

i. clone

j. pull

k. push

l. fetch

m. merge

n. status

o. log

p. checkout

q. Rama / Branch

r. Etiqueta / Tag