

PRÁCTICA DE GIT – GITHUB



El objetivo de la práctica no es ejecutar ningún proyecto como tal, si no practicar GIT, por tanto, puede haber cosas, que, a nivel de desarrollo, no sean coherentes.

ES OBLIGATORIO IR MOSTRANDO “DE VEZ EN CUANDO” EL ESTADO DE LOS FICHEROS MODIFICADOS, ASÍ COMO LOS COMMITS REALIZADOS (AUNQUE NO SE INDIQUE ESPECÍFICAMENTE EN EL DOCUMENTO) Y PONER MENSAJES DESCRIPTIVOS EN CADA COMMIT.

Vamos a partir del Proyecto “Prueba Persona” que deberéis generar utilizando las clases Persona.java y PruebaPersona.java que están disponibles en Classroom.

Realiza los siguientes pasos desde consola:

a) Ubícate en algún lugar de tu disco duro y crea una carpeta llamada **“voluntariagit_tunombre”** (por ejemplo, **voluntariagit_carlos**).

b) Crea en tu equipo local el proyecto PruebaPersona y realiza las acciones necesarias (si las hubiera) para poder empezar a trabajar sobre este proyecto en tu GIT local.

c) Abre el proyecto en tu IDE Eclipse y añade una clase cualquiera al proyecto.

d) Comprueba desde GIT que se han producido cambios en los ficheros del proyecto, y envíalos al Staging y al repositorio local (en dos pasos distintos).

e) Añade comentarios al fichero donde se encuentra la clase del proyecto y realiza un segundo commit al repositorio local.

f) Añade comentarios al fichero de la clase Persona y de nuevo realiza un commit al repositorio local.

g) “Eliminas por error” todos los métodos de la clase Persona, excepto los get-set (si es que se llaman así) y los constructores (si es que hay alguno) y aunque no eres consciente de lo que has borrado, de nuevo haces un commit al repositorio local.

h) Ha sido pulsar <ENTER> y darte cuenta del error (¿¿sudor frío??), pero tu habilidad con GIT hace que todo esté bajo control y solo tengas que volver al commit anterior, por tanto, vuelve al commit anterior y elimina de tu GIT LOCAL el último commit que realizaste (y donde no están la mayoría de los métodos de la clase Persona).

HASTA AQUÍ, TODA LA INFORMACIÓN DE TU PROYECTO DEBE ESTAR EN LA RAMA MASTER O PRINCIPAL

i) Para “curarte en salud” decides enviar todo lo que tienes hasta el momento a tu cuenta de GITHUB. Conecta tu máquina con un repositorio en tu cuenta llamado **“voluntariagit_tunombre”**.

j) Una vez que tienes todo a salvo, decides que te sientes solo desarrollando este proyecto, por tanto, te asocias con un “amigo imaginario” que también controla de repositorios y desarrollo, por lo que necesitas generar una rama adicional en el proyecto. Además, usáis el mismo equipo, ya que vuestros horarios son totalmente distintos. Tu eres más nocturno y tu “amigo” es más bien diurno (o viceversa).

k) La idea es que tu sigas desarrollando en la rama principal y tu “amigo imaginario” en una nueva rama.

l) Crea una nueva rama para tu “amigo” con el nombre que decidas y asegúrate, que justo en este instante, el contenido de la rama principal y de la nueva rama contienen exactamente lo mismo.

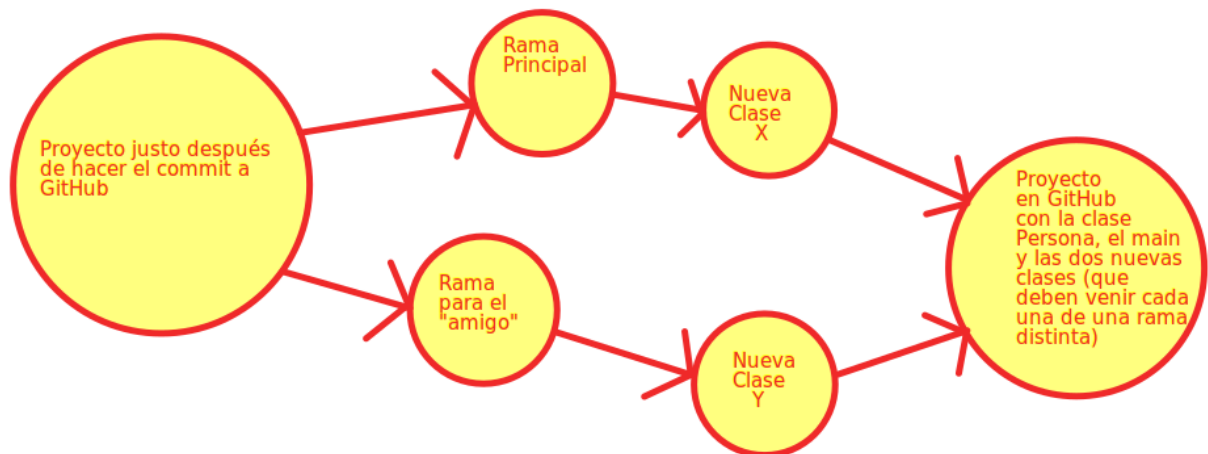
m) En la rama principal añadiremos una nueva clase llamada “claseTuNombre” (añade varias líneas cualesquiera, desde donde quieras, y no es necesario que sea código JAVA).

NOTA: Los ficheros java sabéis que van en la carpeta /src, por tanto, esta clase y la del siguiente apartado deben añadirse ahí.

n) En la nueva rama añadiremos una nueva clase llamada “claseMiAmigoNombre” (añade varias líneas cualesquiera, desde donde quieras, y no es necesario que sea código JAVA).

o) A partir de ahora, el objetivo final es conseguir que en el repositorio de GitHub en la rama master, tengamos toda la parte del proyecto de la clase Persona, y las dos nuevas clases, cada una procedentes de una rama distinta.

Algo como lo que se presenta en la siguiente imagen:



Formato de Entrega: Documento en formato PDF con capturas de pantalla de los pasos, acciones y configuraciones que se van dando para resolver la práctica.

Fecha de Entrega: Miércoles 16 de Febrero (23.59 horas)