Examen de fundamentos de programación

Seccion 1:

1. Todo lo que se halla en el repositorio remoto esta almacenado tanto en la nube de github como también en el almacenamiento interno del ordenador, contrario al local que si bien puede tener nuestro trabajo avanzado, no podría tener, en un caso teorico, todo lo demás que avancemos en otro ordenador, por lo que no importa que dispositivo uses, en el caso del remoto, mientras dichos documentos tengan la ultima versión almacenada en la nube, podras seguir lo que estabas realizando e incluso, guardar dicho progreso y seguir en tu ordenador local sin problemas.
2. El pull request generalmente es usado para verificar o/y comparar códigos en distintas ramas, ya fuesen ramas, o rama y main, y solicitar acciones cuales fuesen commits. Normalmente los conflictos de fusion suceden cuando un mismo documento proveniente de dos ramas se suben a la vez en una misma extensión o rama y coincidentemente se hallan en un mismo sector, renglón o línea, esto genera un merge y para resolverlo el usuario que presenta este merge tiene que separar lo señalado por el sistema, borrarlo o modificarlo.
3. Primeramente realizaría un Fetch para verificar que dichos cambios hallan sido subidos a la nube, si han sido subidos exitosamente, el sistema me pedirá hacer un pull para poder tener esos datos en la versión que tenga de dicho trabajo, realizando un pull, obtendré dicho documento y asi, tanto mi compañero como yo tendremos acceso a la nueva versión del trabajo en cuestión.

Seccion 2:

* Codigo:

int num1, extra, note;

string linea;

extra = 20 / 100;

bool esvalido;

Console.WriteLine("Ingrese la cantidad de respuestas correctas: "); linea = Console.ReadLine();

esvalido = int.TryParse(linea, out num1);

do

{

if (!esvalido)

{

Console.WriteLine("El dato ingresado no es valido, intente de nuevo");

return;

}

else

{

if (num1 >= 8)

{

note = num1 + extra;

Console.WriteLine("La nota obtenida califica como aprobatoria, felicidades.");

Console.WriteLine("Nota calificatoria: " + note);

Console.ReadKey();

}

else if (num1 == 7)

{

note = num1;

Console.WriteLine("Su nota califica como calificatoria:)");

Console.WriteLine("Nota calificada: " + note);

Console.ReadKey();

}

else

{

Console.WriteLine("Su nota no califica como aprobatoria");

Console.WriteLine("Nota calificada: " + num1);

Console.ReadKey();

}

}

} while (!esvalido);

* Diagrama:

Imagen que contiene Escala de tiempo

El contenido generado por IA puede ser incorrecto.

Seccion 3:

1. Codigo: