

Informe TP3 - Reentrega (Changelog)

Correcciones no realizadas

Decidí sacrificar puntos en la nota final en favor de seguir con el TP Final porque estamos bastante atrasados, te pido disculpas por no haber tenido en cuenta dos de las correcciones enviadas.

Las correcciones no realizadas son:

- Algunos strings pasados por copia.
- La forma de parsear comandos no escala, podría haber una entidad que se encargue de este proceso tanto en el cliente como en el servidor. Utilizar polimorfismo.

En cuanto a esta última, aclaro que en el informe de la primer entrega explicito dónde utilicé polimorfismo, según se pide por consigna.

Correcciones realizadas

- En `CertificateHandler Client::_createCertificateFromFile()`, además de los hardcodings, se podría parsear en líneas en un vector de strings y aplicar a cada una la acción correspondiente sin necesidad de contar en un `while`.

Se movió esta operación a un constructor de `CertificateHandler`, se quitó el contador con el `loop` y ahora se utiliza un vector de strings y constantes para determinar en qué posición está cada línea relevante.

- Usar una lista enlazada como contenedor de los `Workers`.

Hecho.

- Poner macros en `.h`.

Todo `#define` fue movido a su respectivo `.h` excepto en casos donde no existe un header file (los `mains` por ejemplo).

- Hardcoding de cantidad y posición de argumentos, y códigos de protocolo.

Se pasó todo esto a constantes.

- Ausencia del return en el main del cliente.

Hecho.

- No está bueno que haya uso de sockets en los main del cliente y servidor, debería tratarse dentro del protocolo.

El uso de sockets se movió a las clases Server y Client, ya no se encuentran en los mains. Me pareció mejor idea mover la inicialización del socket a estas clases, ya que si no se puede bindear/conectar, no tiene sentido proseguir con el resto de la corrida del programa, y creo que esta responsabilidad corresponde tanto al Server como al Client, en lugar de explícitamente al Protocol, el cual lo entiendo como un sistema de reglas para establecer una comunicación.

- Constructor por movimiento de Socket debe hacer lo mismo que el operator= por movimiento.

Hecho.