

Escuela de Ingeniería

Obligatorio de: Programación de Redes

Código de materia: 4864

Fecha: Marzo 2024

Grupo: Todos

Hoja 1 de 2

OBJETIVOS

Poner en práctica los conocimientos adquiridos en la segunda parte del curso.

- Manejo de patrón asincrónico (con `async` / `await`)
- Utilización de librerías de TCP de alto nivel (TCP Listener / TCP Client)

DESCRIPCIÓN DEL SISTEMA: UPGRADE AL SISTEMA TRIPORTUNITY

1. Se debe migrar toda la estructura orientada a Threads **al modelo asincrónico de .NET**. (Esto implica sustituir todos los threads por Tasks y modelar todo en base a Async/Await).
2. Se debe cambiar Sockets por la librería de alto nivel TCPListener y TCP Client.
3. Se debe implementar las mejoras a la entrega anterior especificada por los docentes o que hayan faltado para cumplir los requerimientos de la primera entrega
4. Elegir uno de los siguientes requerimientos e implementarlos:
 - a. Agregar una opción al menú del servidor que muestre el tamaño promedio, tamaño total y cantidad de fotos de los viajes que están en curso. Se debe utilizar algún mecanismo de programación paralela para hacer los cálculos.
 - b. Agregar una opción al servidor que permita cerrar el servidor preguntando previamente si se quiere esperar a que los clientes actuales se desconecten o desconectarlos forzosamente. Se debe aprovechar las funcionalidades de cancelamiento de tareas que trae la clase Task.

REQUERIMIENTOS FUNCIONALES

- Aplicación **Servidor**:
 - o Iguales al obligatorio anterior.
- Aplicación **Cliente**.
 - o Iguales al obligatorio anterior.

ESPECIFICACIÓN DE PROTOCOLO

Pueden seguir utilizando el que habían implementado anteriormente o implementar nuevos cambios.

DOCUMENTACIÓN

Como documentación se espera que se incluya las modificaciones realizadas al código y la motivación de la misma (que beneficios traen sobre el código anterior).

ENTREGA

Escuela de Ingeniería

Obligatorio de: Programación de Redes

Código de materia: 4864

Fecha: Marzo 2024

Grupo: Todos

Hoja 2 de 2

Se debe entregar por Gestión un archivo PDF que contenga la documentación y en la primera página el link al repositorio con el código.

Deben usar el mismo repositorio que la entrega anterior el cual debe pertenecer a la organización de GitHub "ArqSis-PDR-2024-1".

Al realizar la entrega se debe realizar un release en el repositorio y la rama master (o main) no debe ser afectada luego del hito que corresponde a la entrega del obligatorio.

Deben incluir en el repositorio las aplicaciones compiladas en *release*, con ejecutables para Windows (.exe, .dll, etc.) y todo lo necesario para poder ejecutarlas (archivos de configuración, etc.)

Opcionalmente pueden incluir los archivos Dockerfile y docker-compose.yml, junto a una guía, para ejecutar las aplicaciones utilizando Docker.

EVALUACIÓN

El criterio para la corrección se basará en los siguientes aspectos:

- Deberán **evitar el uso de la clase Thread** para el desarrollo del obligatorio. Tampoco estará permitido el uso de binaryformatter para el pasaje de información, se espera que se utilicen streams para hacer esto.
- Por esta entrega no se permitirá que se utilicen parser existentes (como puede ser Json) para serializar objetos.

DEFENSA

La defensa se evaluará junto con la tercera entrega.

Información importante

Lectura de Obligatorio:

Plazo máximo de entrega: 21/5

Puntaje mínimo/máximo: 5/15 puntos.

Los obligatorios se forman como máximo por grupos de 3 estudiantes.

Todas las entregas se realizan en gestión (gestión.ort.edu.uy) y hasta las **21.00** hs del día de entrega.