



CC65 Programación concurrente y distribuida

Enunciado del Trabajo Final del Curso

Profesor:	Carlos Alberto Jara García
Sección:	Todas
Fecha de evaluación:	Semana 15
Ciclo académico:	2024-01

Introducción:

El aprendizaje requiere un estímulo inicial que resulte interesante y nuevo. Precisamente el realizar un análisis del momento actual de la computación, resaltando la evolución de unidades de cómputo en dispositivos con gran capacidad para procesamiento paralelo y conectividad a redes. Así como, buscar la investigación por parte del estudiante de herramientas idóneas para el desarrollo de soluciones y el uso eficaz de los recursos computacionales.

En congruencia con ello, el trabajo final que se propone consiste en la construcción de una solución que implemente una aplicación de programación distribuida y Machine Learning con soporte de Apis, para el problema planteado. El equipo de trabajo utilizará Github como herramienta colaborativa para la gestión del desarrollo de software.

Objetivo:

El presente documento define el trabajo final y la rúbrica que permite evaluar el logro del curso CC65 Programación Concurrente y Distribuida. El objetivo del trabajo final (TF) es que los estudiantes construyan aplicaciones concurrentes y distribuidas de alto rendimiento de manera eficaz desde el punto de vista de la ingeniería de software.

Logro del curso:

Al finalizar el curso, el estudiante construye aplicaciones concurrentes y distribuidas de alto rendimiento de manera eficaz.

El curso busca desarrollar la competencia general de Razonamiento Cuantitativo en nivel 2 y las competencias específicas Responsabilidad y ética en nivel 1 para Ciencias de computación e Ingeniería de Software.

Enunciado

Desarrollar una solución tecnológica para un problema de actualidad que afecta a nuestro país utilizando una arquitectura en capas conformada por una capa UI web (SPA), una capa de APIs contenerizados utilizando docker y una capa cluster conformado por 3 nodos virtualizados como mínimo que ejecute una aplicación que utilice un algoritmo Machine Learning para procesamiento de datos de forma concurrente y distribuida. El equipo de trabajo utilizará Github como herramienta colaborativa para gestionar el código fuente del desarrollo de la aplicación.

Instrucciones:

- El trabajo es en equipos de mínimo 2 y máximo 3 estudiantes, se recomienda mantener los grupos formados en la primera mitad.
- El trabajo consta de dos partes y tendrá como base la TA3 y TA4:
 - La primera parte estará conformada por la revisión bibliográfica, análisis y explicación del algoritmo machine learning a usar en el trabajo.
 - Elaborar el dataset y sustentar su estructura de datos.
 - Cuál es el planteamiento del problema social e indicar la motivación de haber escogido el tema que abordará la solución propuesta.
 - Gráfico de la arquitectura solución propuesta de algoritmos distribuidos cliente/servidor para dividir el procesamiento de la solución en host virtuales mínimo 3 donde la comunicación es a través de puertos y sincronización usando canales y programación concurrente.
 - La segunda parte estará conformada por el desarrollo de la solución propuesta.
- Se usará software para detección de plagio.

Restricciones:

- Para el desarrollo de la interfaz web (SPA) puede utilizar algún framework como Angular, Vue u otro.
- El proyecto se aloja en Github y debe seguir Git Flow. Repositorio público.
- Los dataset deben estar publicados en github y accesibles mediante raw link que será usando en el aplicativo distribuido para procesamiento, el aplicativo no debe de tener localmente el dataset.
- El API debe ser desarrollado íntegramente en GO y contenerizado en Docker.
- El backend de los programas desplegados en el Cluster de host virtuales debe ser desarrollado íntegramente en GO.
- De preferencia usar el algoritmo de machine learning utilizado en la TA4.

Evaluación del Trabajo Final

Instrucciones del entregable:

- La entrega se realizará hasta las 16:00 Hrs. del lunes 24 de junio 2024.
- Elaborar un único informe del trabajo.
- Código fuente de la solución eliminando cualquier defecto.
- Incluir las imágenes y demás recursos utilizados para la elaboración de la solución.
- Video de presentación de la solución en funcionamiento (end to end), realizado en un máximo de 6 minutos donde participen todos los integrantes y demuestren el conocimiento del tema.
- El trabajo será entregado por cada integrante del grupo y mediante el Aula Virtual.
- El plazo es impostergable y por ningún motivo y/o circunstancia se recibirá trabajos fuera de esa fecha y hora, ni por otro medio diferente al aula virtual.

De la entrega

1. El archivo ZIP o RAR a presentar tendrá por nombre TF_XXXXXXXXX, en donde los caracteres X se reemplazarán por el código de alumno.
Este archivo debe contener
 - El informe y el código de su proyecto de aplicación
 - Link al repositorio público de github
 - Link al video subido a un servicio como youtube como “no listado” o similar.
2. Solo se calificarán los trabajos entregados mediante el Aula virtual.
3. Estamos seguros que cada uno realizará su trabajo, sin embargo, para evitar cualquier perspicacia, [le recomendamos leer el reglamento de disciplina del alumno, en el cual se indican las faltas y las sanciones que se indican en el caso de haber copia de trabajos.](#)

El informe debe incluir los siguientes puntos:

- a) Carátula
- b) Resumen
- c) Índice
- d) Descripción del problema y motivación
- e) Objetivos
- f) Sustento del algoritmo utilizado
- g) Explicación del dataset y estructura de datos
- h) Diseño (arquitectura solución)
- i) Desarrollo (evidencia de implementación y pruebas)
- j) Conclusiones
- k) Recomendaciones
- l) Bibliografía (utilizar APA)
- m) Anexos (Link de repositorio GitHub, link de video, documentos, informes, otros)

Rúbrica

	Sobresaliente	Satisfactorio	Deficiente
Planteamiento	3 puntos Usa diagrama de arquitectura solución para complementar la explicación de su planteamiento solución. Programación concurrente y distribuida. Tiene en cuenta las restricciones del trabajo. Cumple con la exposición de su trabajo demostrando conocimiento total del tema.	2 puntos Define procesos e Información incompleta. Expone y no demuestra conocimiento total del tema.	0 puntos No elaborado
Implementación	5 puntos Funciona correctamente, utiliza nodos virtuales, puertos, para procesar los datos generados por la solución concurrentemente, usa canales y muestra resultados siguiendo las consideraciones de la arquitectura solución. Usa Apis para exponer métodos que consume el front end y recopile los resultados del cluster. Tiene en cuenta las restricciones del trabajo. Cumple con la exposición de su trabajo demostrando conocimiento total del tema.	3 puntos Elabora el código y ejecuta concurrentemente, usa puertos, canales, muestra resultados del juego, implementando las consideraciones del trabajo de forma parcial. Expone y no demuestra conocimiento total del tema.	0 puntos No elaborado
Interfaz	5 puntos La interfaz en modo gráfico usando un framework UI, muestra un menú de	3 puntos Muestra el funcionamiento de la aplicación y resultados del juego	0 puntos No elaborado

	<p>opciones, configura el aplicativo con parámetros ingresados desde la UI, muestra los resultados en tiempo real y se apoya mostrando gráficos (tortas, barras, etc) demuestra una adecuada experiencia de usuario y organizada en página SPA. Tiene en cuenta las restricciones del trabajo.</p> <p>Cumple con la exposición de su trabajo demostrando conocimiento total del tema.</p>	<p>entendible en página web parcialmente. Expone y no demuestra conocimiento total del tema.</p>	
Informe	<p>4 puntos</p> <p>Hace buen uso del medio escrito, cumple con la estructura del informe, guarda coherencia y presenta resultados, conclusiones entendibles y orientadas a resultados. Anexa el historial del versionamiento del código fuente, evidencia el git Flow. Cumple con la exposición de su trabajo demostrando conocimiento total del tema.</p>	<p>3 puntos</p> <p>Sigue una estructura clara y contiene los elementos mínimos necesarios solicitados. Usa herramienta de versionado de código. Informe incompleto. Expone y no demuestra conocimiento total del tema.</p>	<p>0 puntos</p> <p>No elaborado</p>
Video	<p>3 puntos</p> <p>Elabora un video de 6 minutos como máximo donde el grupo presenta el desarrollo y funcionamiento del juego, demostrando dominio del tema.</p>	<p>2 puntos</p> <p>Cubre los puntos solicitados de forma parcial.</p>	<p>0 puntos</p> <p>No elaborar el video afecta la calificación en los demás ítems de la rúbrica.</p>

Santiago de Surco, junio de 2024