

Documento Principal (5ptos)

Portada

- Nombre del curso
- Título del proyecto
- Integrantes del equipo
- Nombre del líder
- Fecha de entrega

Índice

1. Introducción

- Presentación general del proyecto
- Justificación del tema elegido
- Objetivos (general y específicos)

2. Descripción del Problema

- Contexto del problema seleccionado
- Aplicación del problema en un escenario real
- Importancia del paralelismo en la solución

3. Cumplimiento de los Requisitos del Proyecto

Para cada criterio, se debe justificar cómo el proyecto lo aborda:

1. Ejecución simultánea de múltiples tareas
2. Necesidad de compartir datos entre tareas
3. Exploración de diferentes estrategias de paralelización
4. Escalabilidad con más recursos
5. Métricas de evaluación del rendimiento
6. Aplicación a un problema del mundo real

4. Diseño de la Solución

- Arquitectura general del sistema
- Diagrama de componentes/tareas paralelas
- Estrategia de paralelización utilizada
- Herramientas y tecnologías empleadas (C#, TPL, etc.)

5. Implementación Técnica

- Descripción de la estructura del proyecto
- Explicación del código clave
- Uso de mecanismos de sincronización

- Justificación técnica de las decisiones tomadas

6. Evaluación de Desempeño

- Comparativa entre ejecución secuencial y paralela
- Métricas: tiempo de ejecución, eficiencia, escalabilidad
- Gráficas o tablas con resultados
- Análisis de cuellos de botella o limitaciones

7. Trabajo en Equipo

- Descripción del reparto de tareas
- Herramientas utilizadas para coordinación (Git, AzureDevops, Trello, Excel, etc.)

8. Conclusiones

- Principales aprendizajes técnicos
- Retos enfrentados y superados
- Posibles mejoras o líneas futuras

9. Referencias

- Fuentes bibliográficas, técnicas o académicas consultadas

10. Anexos

- Manual de ejecución del sistema
- Capturas adicionales, pruebas complementarias
- Enlace al repositorio de Git (público)

Control y Evaluación

Reglas de Entrega en Git (2ptos):

- Repositorio público
- Estructura mínima del repositorio:
 - /docs: Documentación del proyecto
 - /src: Código fuente
 - /tests: Pruebas
 - /metrics: Resultados de comparativas realizadas
- **Cada miembro debe:**
 - Subir al menos 3 *commits significativos* (implementación o documentación) en las semanas de clases.

- Usar su nombre en los *commits*, identificar con `Co-authored-by-<nombreApellidoEnPascalCase>`

Evaluación del trabajo en equipo (3ptos)

Cada elección debe ser una escala entre: No satisfactorio, Satisfactorio, Bueno, Muy bueno

Por el líder:

- Evalúa a cada miembro en 3 aspectos:
 - Compromiso y responsabilidad
 - Calidad de su código / contribuciones
 - Colaboración y comunicación

Entre miembros:

- Cada miembro evalúa a los otros según:
 - Ayuda en resolución de problemas
 - Participación en decisiones
 - Claridad en la comunicación

Miembros al líder:

- Evalúan en:
 - Organización y delegación de tareas
 - Resolución de conflictos
 - Apoyo técnico y emocional

Exposición. 10ptos.

Criterio	Descripción	Puntos
1. Claridad en la explicación del problema (tema elegido)	El equipo explica claramente el contexto, la importancia del problema y por qué fue elegido.	1
2. Dominio técnico del tema	Se demuestra comprensión profunda del paralelismo, sincronización, y estrategias aplicadas en su solución.	1
3. Presentación del diseño y arquitectura	El diseño está bien explicado, visualmente apoyado (diagrama), y se entienden las decisiones técnicas.	1
4. Demostración del funcionamiento del sistema	El código se muestra funcionando, se explican los fragmentos relevantes y hay claridad en la ejecución.	5

Criterio	Descripción	Puntos
5. Interpretación de resultados y métricas	Se explican los resultados de forma clara con comparativas significativas (secuencial vs paralela).	1
6. Gestión del tiempo	La exposición se ajusta al tiempo máximo sin omitir ni extender innecesariamente.	1

El líder del equipo debe asegurar el cumplimiento de cada sección descrita en este documento.