

PROYECTO INGENIERÍA DE SOFTWARE IV

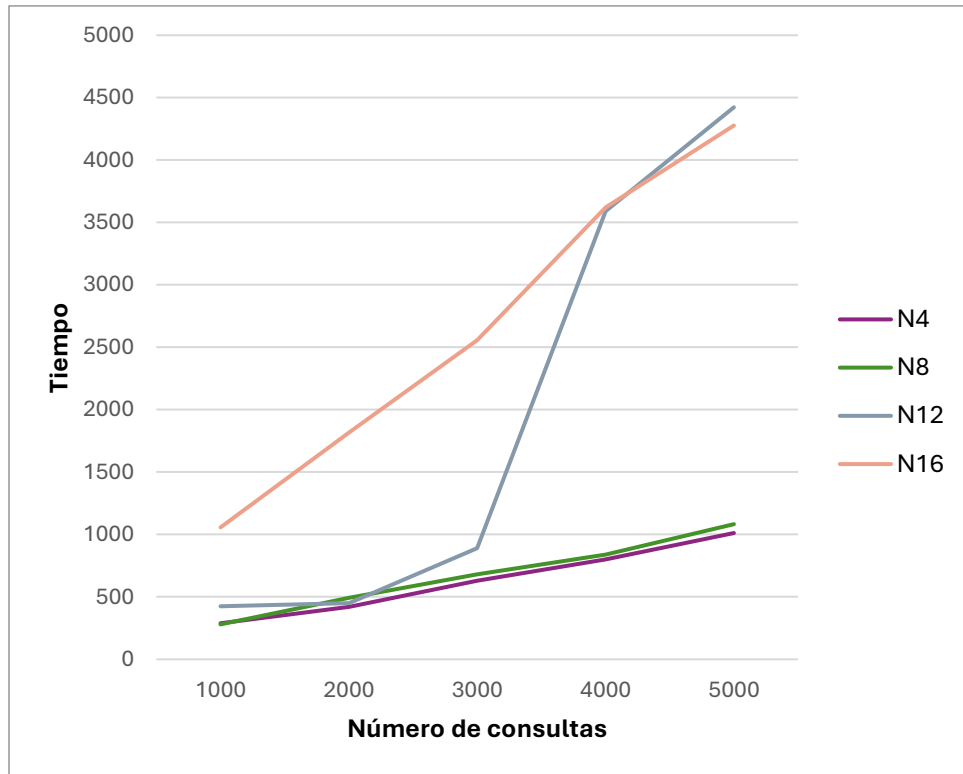
Diseño de Experimentos del Monolítico

Parte 1: Comparación de Diferentes Tamaños de Pool

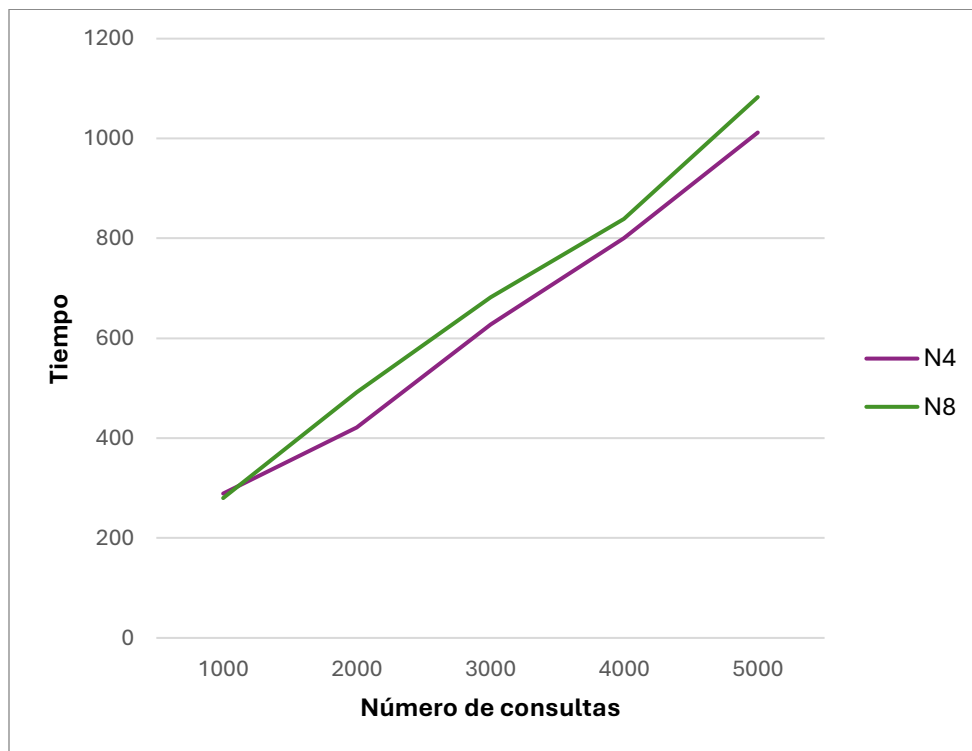
En esta parte del experimento, se compararon diferentes tamaños de pool (4, 8, 12 y 16) para determinar cuál es el más adecuado para usar en la distribución. Se realizaron varias ejecuciones con diferentes tamaños de pool y se midió el tiempo de ejecución para diferentes cantidades de consultas. En la hoja “[Monolito-Datos](#)” del archivo Excel encontrará las ejecuciones realizadas para esta parte.

Número de pool	Tamaño	Tiempo (ms)
4	1000	289
	2000	421
	3000	628
	4000	801
	5000	1012
8	1000	280
	2000	492
	3000	682
	4000	839
	5000	1083
12	1000	425
	2000	450
	3000	892
	4000	3592
	5000	4423
16	1000	1057
	2000	1816
	3000	2558
	4000	3623
	5000	4277

Tabla 1.



Gráfica 1. Interacción número de consultas vs tiempo para los tamaños de pool



Gráfica 2. Interacción detallada para los pools de tamaño 4 y 8

Con estos resultados, el pool de 4 hilos demuestra ser el más eficiente en términos de tiempo de ejecución para tamaños de consulta de hasta 5000. El pool de 8 hilos también es una opción viable, pero no ofrece una mejora significativa en comparación con el pool de 4 hilos. Los pools de 12 y 16 hilos muestran una sobrecarga considerable y no son recomendables para este escenario. Con esto podemos concluir, que un menor número de hilos puede ser más eficiente debido a la menor sobrecarga de cambio de contexto y contención de recursos para los tamaños analizados.

Parte 2: Determinación del Número de Consultas Cercanas a 1 Segundo

En esta parte del experimento, se tomó como referencia un pool de 16 hilos para encontrar el número de consultas que se acercan más a 1 segundo de tiempo de ejecución, esto debido a que el computador analizado cuenta con 16 hilos virtuales. Luego, con esa cantidad de consultas, se realizaron ejecuciones con diferentes tamaños de pool para ver cuál beneficia más. En la hoja [“Exp Monolito”](#) del archivo Excel encontrará las ejecuciones realizadas para esta parte.

Número de Hilos para el Pool	Total de Consultas	Tiempo Total de Ejecución (ms)
16	4870	1090
4	4870	935
8	4870	979
12	4870	949

Tabla 2.

El pool de 4 hilos resultó ser el más eficiente, con un tiempo total de ejecución de 935 ms para 4870 consultas. Esto sugiere que, para este número de consultas, un pool de 4 hilos proporciona el mejor rendimiento en términos de tiempo de ejecución. Por otro lado, para el pool de 16 hilos, aunque se esperaba que un mayor número de hilos mejorara el rendimiento, la sobrecarga de manejo de hilos adicionales parece haber afectado negativamente el tiempo de ejecución de manera no muy significativa. El método de tanteo permitió ajustar las configuraciones y encontrar la combinación óptima de número de consultas y tamaño del pool

Método de Tanteo:

El método de tanteo se utilizó para ajustar el número de consultas y el tamaño del pool para encontrar la configuración que se acerque más a 1 segundo de tiempo de ejecución. Se realizaron múltiples ejecuciones con diferentes configuraciones y se midieron los tiempos

de ejecución para determinar la configuración óptima. Este método implica probar diferentes combinaciones de parámetros y observar los resultados para identificar la mejor configuración.

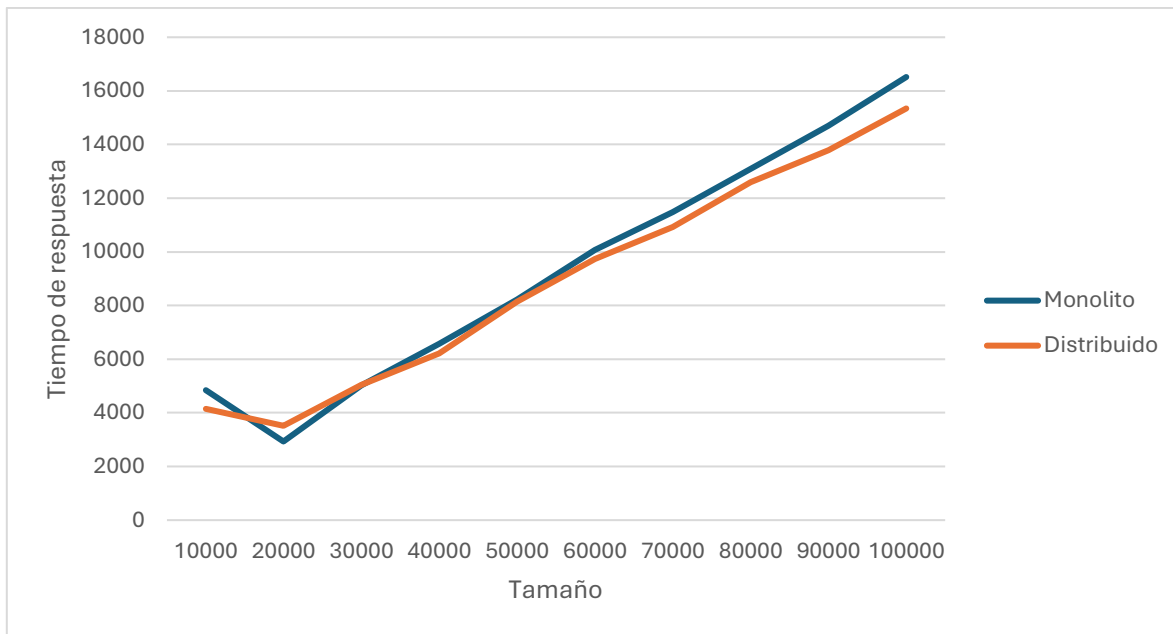
Diseño de Experimentos Distribuido

En este experimento, se compararon los tiempos de respuesta de un sistema monolítico y un sistema distribuido para diferentes tamaños de consultas. El objetivo es determinar cómo se comportan ambos sistemas a medida que aumenta el número de consultas y cuál de ellos ofrece un mejor rendimiento en términos de tiempo de respuesta.

Se realizaron ejecuciones con tamaños de consultas que van desde 10,000 hasta 100,000, incrementando de 10,000 en 10,000. Para cada tamaño de consulta, se midió el tiempo de respuesta tanto del sistema monolítico como del sistema distribuido. En la hoja [“Monolito-Datos”](#) y [“Distribuido-Datos”](#) del archivo Excel encontrará las ejecuciones realizadas para esta parte. Los resultados obtenidos se presentan en la siguiente tabla:

Tamaño	Tiempo de Respuesta Monolito (ms)	Tiempo de Respuesta Distribuido (ms)
10000	4848	4146
20000	2931	3517
30000	5026	5047
40000	6573	6213
50000	8230	8153
60000	10073	9731
70000	11493	10930
80000	13093	12602
90000	14704	13787
100000	16518	15346

Tabla 3.



Gráfica 3.

Comparación de Tiempos de Respuesta

Para 10,000 consultas, el sistema distribuido mostró un mejor rendimiento con un tiempo de respuesta de 4146 ms, en comparación con el sistema monolítico que tuvo un tiempo de respuesta de 4848 ms. Esto sugiere que el sistema distribuido puede manejar eficientemente una carga de trabajo moderada, reduciendo el tiempo de respuesta en comparación con el sistema monolítico.

Sin embargo, para 20,000 consultas, el sistema monolítico tuvo un mejor rendimiento con un tiempo de respuesta de 2931 ms, mientras que el sistema distribuido tuvo un tiempo de respuesta de 3517 ms. Este resultado indica que, en ciertos escenarios, el sistema monolítico puede ser más eficiente, posiblemente debido a la menor sobrecarga de comunicación entre componentes en un entorno monolítico.

A medida que el tamaño de las consultas aumenta a 30,000, ambos sistemas muestran tiempos de respuesta muy similares, con 5026 ms para el monolítico y 5047 ms para el distribuido. Esto indica que ambos sistemas manejan este tamaño de consulta de manera comparable, sin una ventaja clara para ninguno de los dos.

Para 40,000 consultas, el sistema distribuido vuelve a mostrar un mejor rendimiento con un tiempo de respuesta de 6213 ms, en comparación con 6573 ms del sistema monolítico. Este patrón se mantiene para tamaños de consulta mayores, donde el sistema distribuido generalmente ofrece tiempos de respuesta más bajos. Por ejemplo, para 50,000 consultas, el tiempo de respuesta del sistema distribuido es de 8153 ms, mientras que el del monolítico es de 8230 ms.

Tendencias Observadas

A medida que el tamaño de las consultas aumenta, el sistema distribuido muestra una mejor escalabilidad y tiempos de respuesta más bajos en comparación con el sistema monolítico. Esto sugiere que el sistema distribuido es más eficiente para manejar grandes volúmenes de consultas, beneficiándose de la distribución de la carga de trabajo entre múltiples nodos.

Sin embargo, hay variabilidad en los resultados, como se observa con 20,000 consultas, donde el sistema monolítico tiene un mejor rendimiento. Esto indica que la eficiencia puede variar dependiendo de la carga de trabajo específica y la configuración del sistema. Es posible que, en ciertos escenarios, la simplicidad y menor sobrecarga del sistema monolítico ofrezcan ventajas en términos de rendimiento.