Diego Ortiz 201224923

Juan Felipe Calderon 201226725

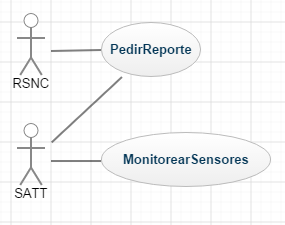
Andres Verdugo 201118331

Paula Ramirez 201328023

Experimento 1 – Documentación

En el experimento 1 se utilizó una arquitectura que cumpla con atributos de calidad de desempeño y escalabilidad que se plantean más abajo. En el desarrollo del software se realizaron capas de servicios y lógica para contribuir a la escalabilidad del proyecto. Así mismo, con respecto a los servicios REST, se hizo uso de Threads asincrónicos con el fin de contribuir al desempeño del programa, es decir, el sistema después de recibir la información de alerta genera estos threads para buscar el sensor mas cercano al evento. La arquitectura REST permite que la aplicación sea escalable a futuro ya que su estructura es de forma desacoplada lo que permite fácilmente que se adicionen mas funciones y clases al proyecto sin necesidad de hacer mayores cambios en el. De igual manera el protocolo de respuesta de REST (en url) hace que sea más fácil de manipular que otras arquitecturas y contribuye al procesamiento más rápido de las peticiones y respuestas web haciendo que el desempeño sea mejor.

**Casos de uso:**



**Escenarios de calidad:**

**Escenario 1**

**Desempeño**

|  |  |
| --- | --- |
| **Indicador** | **Prioridad** |
| ES1 | Alta |
| **Fuente** | |
| SATT | |
| **Estimulo** | |
| Reporte de un evento por parte de RSNC | |
| **Ambiente** | |
| Normal | |
| **Medida esperada** | |
| Se envía reporte a la RSNC del sensor más cercano al sitio del evento en menos de 500 ms | |

**Escenario 2**

**Desempeño**

|  |  |
| --- | --- |
| **Indicador** | **Prioridad** |
| ES2 | Media |
| **Fuente** | |
| Sensor | |
| **Estimulo** | |
| Hacer seguimiento de las alarmas | |
| **Ambiente** | |
| Normal | |
| **Medida esperada** | |
| Sensores envían los datos medidos cada 60 segundos. | |

**Escenario 3**

**Escalabilidad**

|  |  |
| --- | --- |
| **Indicador** | **Prioridad** |
| ES3 | Media |
| **Fuente** | |
| Sensor | |
| **Estimulo** | |
| Un cambio de más de 1.5 m en una ola de la zona costera más cercana al sitio del evento | |
| **Ambiente** | |
| Normal | |
| **Medida esperada** | |
| Se espera que el sistema reciba la alerta de un sensor y siga recibiendo información del resto de sensores | |

**Escenario 4**

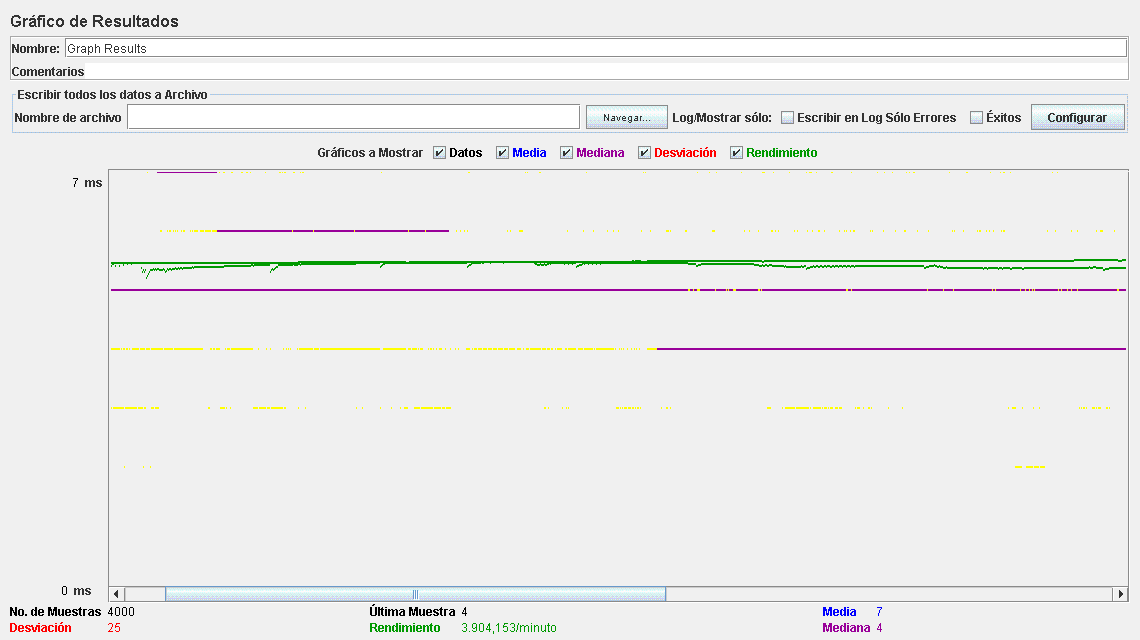
**Escalabilidad**

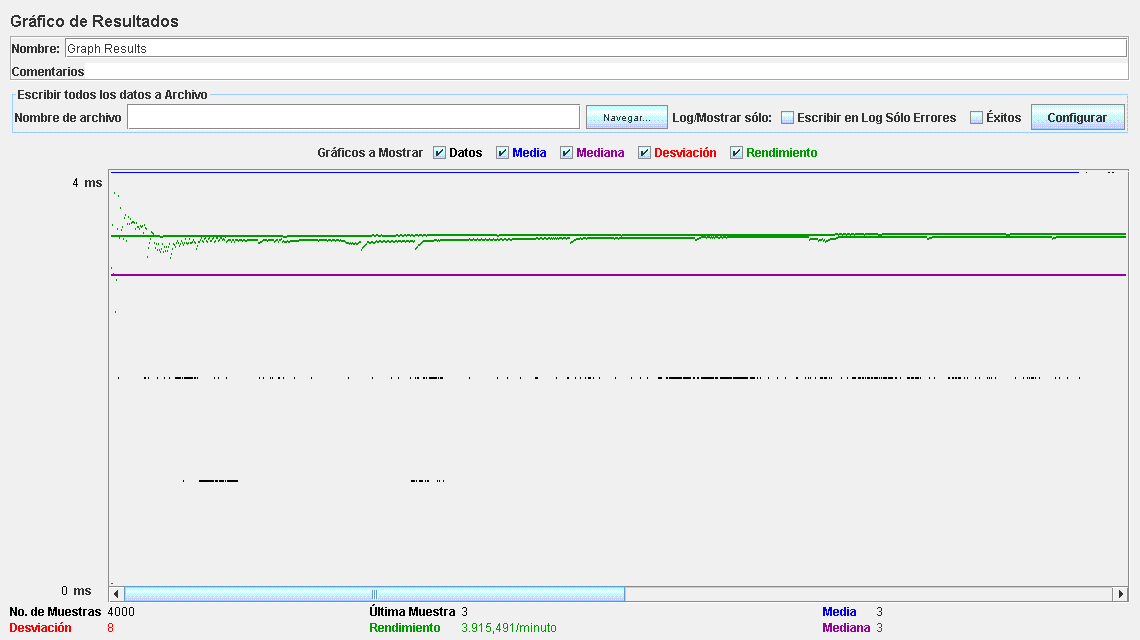
|  |  |
| --- | --- |
| **Indicador** | **Prioridad** |
| ES4 | Media |
| **Fuente** | |
| Sensor | |
| **Estimulo** | |
| Necesidad de agregar más sensores para un monitoreo más exhaustivo | |
| **Ambiente** | |
| Normal | |
| **Medida esperada** | |
| Se espera que el sistema pueda recibir información de más sensores de los inicialmente instalados | |

**Métricas:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Escenario de calidad** | **Métrica** | **Valor esperado** | **Valor obtenido** |
| **ES1** | Latencia | 500ms | 20-25 ms |
| **ES2** | Número de peticiones respondidas en un minuto | 4000 | 3910 |
| **ES3** | Tiempo de llegada de una alerta de un sensor | 500ms |  |
| **ES4** | Tiempo de recibir información de todos los sensores | 60s |  |

**Resultados pruebas de carga:**





El servidor fue capaz de responder aproximadamente a 3910 solicitudes de los sensores hechas en 1 minuto lo que refleja deficiencias en el código asociado ya que este recorre todos los sensores que se encuentran hasta encontrar el que va a ser actualizado por lo cual debe mejorarse.

Generador de Reporte

El reporte se genera aproximadamente entre 20 y 25 milisegundos, en las pruebas efectuadas se dieron tiempos de 31.2ms, 25ms, 19.01ms, y 21.97ms. Esto en procesar la información dada por la RSCN y generar el reporte devuelta indicando la altura y la alerta necesaria.