Sistema de Seguimiento Cardiaco: Disponibilidad

Grupo 11

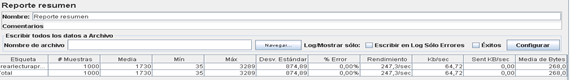
Andrea Juliana Paredes

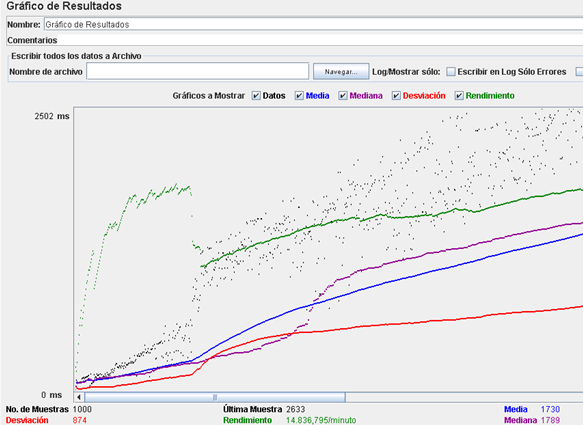
Cristian Camilo Novoa

Universidad de los Andes, Bogotá, Colombia  
Ingeniería de Sistemas y Computación   
Diseño y Arquitectura de Software  
2017

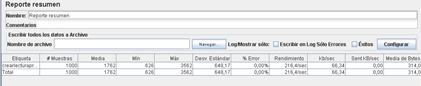
**Contenido**

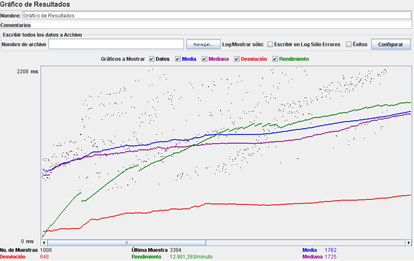
1. **Pruebas de Carga**
   1. **Pruebas de Carga Experimento Actual**
   2. **Pruebas de Carga Experimento Anterior**
   3. **Análisis y Comparación**
2. **Explicación de Decisiones**
3. **Pruebas de Carga**
   1. **Pruebas de Carga Experimento Anterior**





* 1. **Pruebas de Carga Experimento Actual**

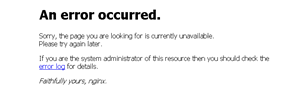
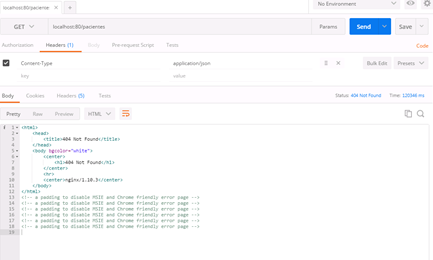




* 1. **Análisis y Comparación**

Cómo se puede observar en las gráficas, se implementó un balanceador de carga el cual en primera oportunidad mejora la disponibilidad y la estabilidad de la aplicación en el momento de carga, esto se puede evidenciar comparando el valor de la desviación en ambas gráficas, al implementar el balanceador así como beneficia unos atributos de calidad, también afecta negativamente otros como es el caso del desempeño que se ve afectado causando un aumento en el tiempo de respuesta en la carga de datos, debido al procesamiento adicional que se hace para distribuir los datos entre los servidores

1. **Explicación de Decisiones**

En este experimento los cambios que se hicieron fueron un balanceador de carga, que se encarga de administrar las peticiones entrantes y repartirlas en los distintos servidores activos que haya, para esto se usaron varios nodos de computación y replicación de datos permitiendo así la mejoría de la disponibilidad del programa, dado que en caso de algún fallo en un servidor el balanceador deja de enviar peticiones a ese y las reparte entre los otros, así es mucho más difícil que la aplicación no esté disponible en algún momento. En caso de que todos los servidores fallaran se presentan los errores mostrados a continuación:  
  
  
Con seguridad

Se agregó una clase encriptadora para cifrar y descifrar las lecturas enviadas desde el microcontrolador que implementa los sensores. Esto se hizo con el objetivo de cumplir el atributo de integridad de datos.

Se puede evidenciar que la media con respecto al experimento anterior aumenta, esto es debido a que el encriptador actúa como un intermediario mientras encripta y desencinta las lecturas enviadas. También hubo un pequeño porcentaje de error de algunas lecturas que no fueron procesadas debido a que el sistema se encontraba ocupado (tiempo de cifrado y descifrado).

Se intentó implementar un sistema de autenticación y autorización, pero al final no fue posible por desconocimiento de la tecnología.

