

### **Experimento 3.**

#### **Técnicas utilizadas:**

Para la interfaz o Front-End se utilizaron cuatro diferentes tácticas a razón de distintas necesidades y limitaciones del sistemas. En comparación con lo que se tenía en mente en los Mockups hay ciertas diferencias; estas serán explicadas y argumentadas más adelante.

#### **Estructurar las interfaces en función de tareas:**

La primera entrada del servicio es la entrada de usuario, después de esto se hace la presentación por medio de pestañas, en estas pestañas se encuentran cada una de las tareas que se ofrecen.

Se decidió hacer de esta forma pues consideramos que es la forma más sencilla y concisa de hacer tal presentación, así solo debe dirigirse directamente a lo que quiere conocer hacer una entrada de búsqueda en los casos que sea necesario y obtener la información necesaria.

#### **Ayudar a navegar las interfaces:**

Como característica principal dejamos y de mayor importancia fue la forma en que el usuario puede hacer uso de las diferentes funciones que ofrece el sistema. Esto se debe a que en un caso de uso real no solo no se tiene el tiempo para hacer tareas tediosas o que simplemente no permitan el menor tiempo a los resultados requerido; esto afectaría la usabilidad y de esta forma ser menos atractivo para los clientes.

Para esto se tiene una presentación sencilla y clara. En la parte superior hay pestañas desde las cuales se puede acceder a las distintas funciones y servicios que se ofrecen, dentro de estas está el caso en que se pueda hacer una búsqueda por zona y área, la cual retorna todos los valores de tal; estos también pueden ser alerta. Por otro lado se puede ver lecturas en tiempo real de las distintas mediciones (CO2, Iluminación, Temperatura, Sonido).

De esta forma acceder a las distintas funciones sólo es cuestión de acceder con el usuario determinado y hacer la selección deseada.

#### **Mantener informado al usuario de su ubicación:**

A pesar de que las funciones están bien definidas y son de fácil acceso se el usuario puede ver en donde se encuentra en términos de función y si es el caso de su búsqueda.

De esta forma si se presenta el caso que el usuario se aparta por un momento determinado del sistema al volver puede saber fácilmente donde se encuentra y qué información está viendo. Así se evita tener que volver a hacer búsquedas o simplemente ver una gran cantidad de información sin tener un contexto que permita entenderlas.

#### **Dashboards compactos y modulares:**

Los dashboards son la presentación de la información de forma más gráfica. Esto ayuda a que se ahorre tiempo al momento de querer “entenderla” pues de esta forma ya se tienen marcos de referencia y comparación. Por otro lado decidimos que fuera lo más sencillo posible y que solo mostrará la información necesaria para que se pudieran tomar decisiones rápidas y relativamente confiables.

### **MockUp vs Entrega:**

En comparación con los MockUps que se habían entregado en entregas pasadas, se puede ver que hay un cambio.

La forma en que nosotros habíamos diseñado el Front-End hacía un resumen de las diferentes mediciones, alertas y mostraba la información en vivo todo en un mismo lugar, de forma compacta y lo más resumida posible; de todas formas se permitía conocer más sobre cada una de estas haciendo click en su respectiva representación.

A pesar de que para nosotros era la mejor forma de hacer la presentación después de presentarlo se nos comunicó que esto podría hacer que el usuario se sintiera abrumado por tanta información que se le era presentado lo cual incluso podría ser más una desventaja del producto, por tal preferimos dividir los diferentes servicios en diferentes interfaces que permiten una presentación sencilla y ayuda al usuario a que solo obtenga la información que quiere conocer.

### **Patrones de diseño**

En el proyecto se implementaron tres diferentes tipos de patrones de diseño, Wrapper, Proxy, y Facade. El wrapper se implementó a través de las clases de converter, las cuales reciben por un lado entities tanto en lista como individuales y los convierte en DTOs y también ejecuta el proceso contrario. Esto es necesario ya que la persistencia solamente recibe entities y la lógica utiliza DTOs. El proxy se implementó en la clase persister, la cual recibe objetos genéricos y se encarga de persistirlos en la base de datos. Esto permite que cualquier clase de lógica se comunique con la base de datos sin problemas y sin implementar sus propios métodos. Por último está el facade, el cual se implementó en

## Comparación de resultados

### Resultados esperados:

Procesar hasta 5000 peticiones con 0% de error y tiempos de respuesta inferiores a 30 milisegundos

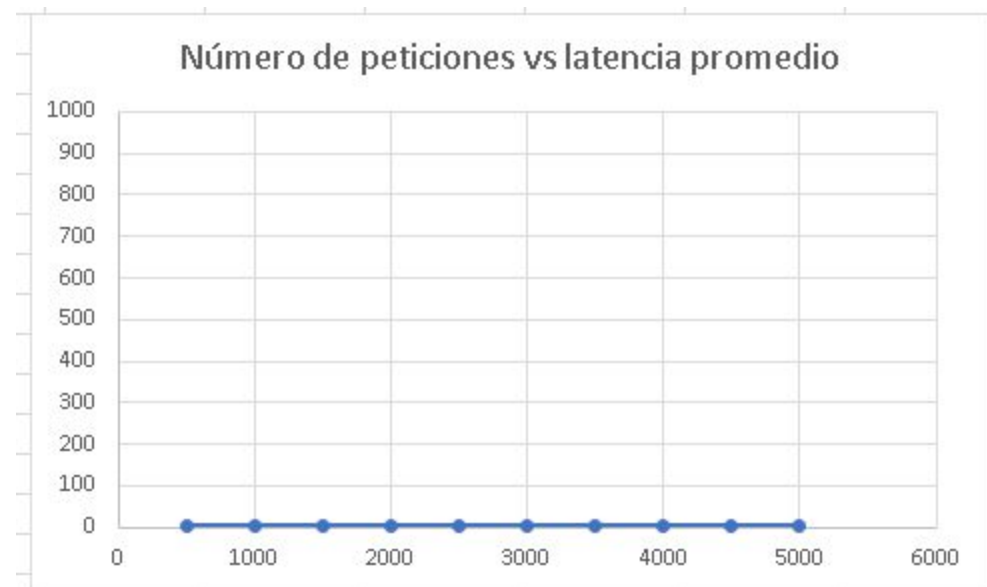
### Resultados obtenidos:

Ejecutamos las pruebas en Jmeter primero con 2500 peticiones y luego con 5000. Obtuvimos 0% de error en ambas y tiempo de respuesta promedio de 4 milisegundos.

A continuación se presenta el registro de resultados obtenidos:

alertas					
Muestras	Media	Min	Max	%Error	Rendimiento()
500	4	3	17	0	8.3
1000	4	3	23	0	16.5
1500	4	2	11	0	24.7
2000	4	2	10	0	32.7
2500	4	2	26	0	40.7
3000	4	2	73	0	48.3
3500	4	2	109	0	56.3
4000	4	2	199	0	63.8
4500	4	2	85	0	72.2
5000	4	2	139	0	79.1

### Reporte gráfico:





Pantallazos pruebas:

500

Reporte resumen											
Nombre: Summary Report											
Comentarios											
Escribir todos los datos a Archivo											
Nombre de archivo				Navegar...	LogMostrar sólo:			<input type="checkbox"/> Escribir en Log Sólo Errores	<input type="checkbox"/> Éxitos	Configurar	
Etiqueta	# Muestras	Media	Min	Máx	Desv. Estánd.	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt.	
Write - Add R...	500	4	3	17	1,18	0,00%	8,3/sec	2,37	1,84	292,0	
Total	500	4	3	17	1,18	0,00%	8,3/sec	2,37	1,84	292,0	

1000

Reporte resumen											
Nombre: Summary Report											
Comentarios											
Escribir todos los datos a Archivo											
Nombre de archivo				Navegar...	LogMostrar sólo:			<input type="checkbox"/> Escribir en Log Sólo Errores	<input type="checkbox"/> Éxitos	Configurar	
Etiqueta	# Muestras	Media	Min	Máx	Desv. Estánd.	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt.	
Write - Add R...	1000	4	3	23	1,20	0,00%	16,5/sec	4,71	3,66	292,0	
Total	1000	4	3	23	1,20	0,00%	16,5/sec	4,71	3,66	292,0	

1500

Reporte resumen											
Nombre: Summary Report											
Comentarios											
Escribir todos los datos a Archivo											
Nombre de archivo				Navegar...	LogMostrar sólo:			<input type="checkbox"/> Escribir en Log Sólo Errores	<input type="checkbox"/> Éxitos	Configurar	
Etiqueta	# Muestras	Media	Min	Máx	Desv. Estánd.	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt.	
Write - Add R...	1500	4	2	11	0,90	0,00%	24,7/sec	7,04	5,47	292,0	
Total	1500	4	2	11	0,90	0,00%	24,7/sec	7,04	5,47	292,0	

2000

**Reporte resumen**

Nombre: Summary Report

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo   Log/Mostrar sólo: ☐ Escribir en Log ☐ Sólo Errores ☐ Éxitos

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estánd.	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt.
Write - Add R..	2000	4	2	10	0,95	0,00%	32,7/sec	9,33	7,26	292,0
Total	2000	4	2	10	0,95	0,00%	32,7/sec	9,33	7,26	292,0

2500

**Reporte resumen**

Nombre: Summary Report

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo   Log/Mostrar sólo: ☐ Escribir en Log ☐ Sólo Errores ☐ Éxitos

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estánd.	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt.
Write - Add R..	2500	4	2	26	1,79	0,00%	40,7/sec	11,60	9,02	292,0
Total	2500	4	2	26	1,79	0,00%	40,7/sec	11,60	9,02	292,0

3000

**Reporte resumen**

Nombre: Summary Report

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo   Log/Mostrar sólo: ☐ Escribir en Log ☐ Sólo Errores ☐ Éxitos

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estánd.	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt.
Write - Add R..	3000	4	2	73	1,68	0,00%	48,3/sec	13,77	10,71	292,0
Total	3000	4	2	73	1,68	0,00%	48,3/sec	13,77	10,71	292,0

3500

**Reporte resumen**

Nombre: Summary Report

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo   Log/Mostrar sólo: ☐ Escribir en Log ☐ Sólo Errores ☐ Éxitos

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estánd.	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt.
Write - Add R..	3500	4	2	109	2,78	0,00%	56,3/sec	16,07	12,49	292,0
Total	3500	4	2	109	2,78	0,00%	56,3/sec	16,07	12,49	292,0

4000

**Reporte resumen**

Nombre: Summary Report

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo   Log/Mostrar sólo: ☐ Escribir en Log ☐ Sólo Errores ☐ Éxitos

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estánd.	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt.
Write - Add R..	4000	4	2	199	4,73	0,00%	63,8/sec	18,20	14,15	292,0
Total	4000	4	2	199	4,73	0,00%	63,8/sec	18,20	14,15	292,0

4500

**Reporte resumen**

Nombre:

Comentarios:

Escribir todos los datos a Archivo

Nombre de archivo:   LogMostrar sólo: ☐ Escribir en Log Sólo Errores ☐ Éxitos

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estánd.	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt.
Write - Add R...	4500	4	2	85	4,36	0,00%	72,2/sec	20,57	15,99	292,0
Total	4500	4	2	85	4,36	0,00%	72,2/sec	20,57	15,99	292,0

5000

**Reporte resumen**

Nombre:

Comentarios:

Escribir todos los datos a Archivo

Nombre de archivo:   LogMostrar sólo: ☐ Escribir en Log Sólo Errores ☐ Éxitos

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estánd.	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Byt.
Write - Add R...	5000	4	2	139	6,24	0,00%	79,1/sec	22,56	17,54	292,0
Total	5000	4	2	139	6,24	0,00%	79,1/sec	22,56	17,54	292,0