



Testing report

C2.034

<https://github.com/juanlo017/c1.034>

Apellidos, Nombre	Email
Castro Vázquez, Jose Ignacio	joscasvaz@alum.us.es

Tabla de contenidos

Resumen ejecutivo	3
Tabla de revisiones	4
Introducción	5
Contenidos	6
Testing Funcional	6
Contract (Client)	6
Progress Logs (Client)	15
Testing de rendimiento	22
Conclusiones	27
Bibliografía	28

Resumen ejecutivo

En este documento se detalla todo el proceso de testing formal de los requisitos #6 y #7 del Student #2.

Tabla de revisiones

Versión	Descripción	Fecha
v1.0	Creación del documento	14/05/2024
v1.1	Versión final	06/07/2024

Introducción

Se ha procedido a testear formalmente los requisitos #6 y #7 del Student #2.

Contenidos

Testing Funcional

En esta sección se listan los casos de prueba de las funcionalidades con una descripción de cómo se han llevado a cabo y si han sido efectivos a la hora de encontrar bugs.

Contract (Client)

Funcionalidad	Descripción	Bugs
LIST-SAFE	Navegar como usuario con rol client a la url, comprobar que efectivamente se muestra la vista y no lanza ningún error el sistema.	No se ha encontrado ningún bug.
LIST-HACK	Navegar como usuario sin rol o rol distinto a client a la url, comprobar que efectivamente no muestra la vista y que lanza error el sistema.	No se ha encontrado ningún bug.
SHOW-SAFE	Navegar como usuario con rol client y dueño de la entidad, comprobar que efectivamente se muestra la vista y no lanza ningún error el sistema.	No se ha encontrado ningún bug.

SHOW-HACK	Navegar como usuario sin rol client o con un client ajeno a la entidad a la url, comprobar que efectivamente no se muestra la vista y que lanza error el sistema.	No se ha encontrado ningún bug.
-----------	---	---------------------------------

<p>CREATE-SAFE</p>	<ul style="list-style-type: none"> - Navegar como usuario con rol client y dueño de la entidad, comprobar que efectivamente se muestra la vista y no lanza ningún error el sistema. - Probar todos los inputs del formulario de acuerdo con la metodología dada en clase: <p>CODE:</p> <ul style="list-style-type: none"> Probar sin código Probar código duplicado en la bddd Probar sin letras Probar sin guión Probar sin número Probar con menos números Probar con más números <p>PROVIDER NAME:</p> <ul style="list-style-type: none"> Probar vacío Probar código html Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter <p>CUSTOMER NAME:</p> <ul style="list-style-type: none"> Probar vacío Probar código html Probar código de lógica 	<p>No se ha encontrado ningún bug.</p>
--------------------	--	--

	<p>Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter</p> <p>GOALS:</p> <p>Probar vacío Probar código html Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter</p> <p>BUDGET:</p> <p>Probar sin budget Probar budget negativo Probar con 0 Probar con 1.01 Probar con el máximo, que es el budget que falta para alcanzar el coste del proyecto</p> <p>Probar con una divisa distinta a la del coste del proyecto</p> <p>PROJECT:</p> <p>Probar vacío Probar con un proyecto del desplegable</p>	
--	---	--

UPDATE-SAFE	<ul style="list-style-type: none"> - Navegar como usuario con rol client y dueño de la entidad, comprobar que efectivamente se muestra la vista y no lanza ningún error el sistema. - Probar todos los inputs del formulario de acuerdo con la metodología dada en clase: <p>CODE:</p> <ul style="list-style-type: none"> Probar sin código Probar código duplicado en la bddd Probar sin letras Probar sin guión Probar sin número Probar con menos números Probar con más números Probar con el antiguo código <p>PROVIDER NAME:</p> <ul style="list-style-type: none"> Probar vacío Probar código html Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter <p>CUSTOMER NAME:</p> <ul style="list-style-type: none"> Probar vacío Probar código html 	No se ha encontrado ningún bug.
-------------	---	---------------------------------

	<p>Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter</p> <p>GOALS:</p> <p>Probar vacío Probar código html Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter</p> <p>BUDGET:</p> <p>Probar sin budget Probar budget negativo Probar con 0 Probar con 1.01 Probar con el máximo, que es el budget que falta para alcanzar el coste del proyecto</p> <p>Probar con una divisa distinta a la del coste del proyecto</p> <p>PROJECT:</p> <p>Probar vacío Probar con un proyecto del desplegable</p>	
--	--	--

PUBLISH-SAFE	<ul style="list-style-type: none"> - Navegar como usuario con rol client y dueño de la entidad, comprobar que efectivamente se muestra la vista y no lanza ningún error el sistema. - Probar todos los inputs del formulario de acuerdo con la metodología dada en clase: <p>CODE:</p> <ul style="list-style-type: none"> Probar sin código Probar código duplicado en la bddd Probar sin letras Probar sin guión Probar sin número Probar con menos números Probar con más números Probar con el antiguo código <p>PROVIDER NAME:</p> <ul style="list-style-type: none"> Probar vacío Probar código html Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter <p>CUSTOMER NAME:</p> <ul style="list-style-type: none"> Probar vacío Probar código html 	No se ha encontrado ningún bug.
--------------	---	---------------------------------

	<p>Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter</p> <p>GOALS:</p> <p>Probar vacío Probar código html Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter</p> <p>BUDGET:</p> <p>Probar sin budget Probar budget negativo Probar con 0 Probar con 1.01 Probar con el máximo, que es el budget que falta para alcanzar el coste del proyecto</p> <p>Probar con una divisa distinta a la del coste del proyecto</p> <p>PROJECT:</p> <p>Probar vacío Probar con un proyecto del desplegable</p>	
--	--	--

DELETE-SAFE		No se ha encontrado ningún bug.
-------------	--	---------------------------------

Progress Logs (Client)

Funcionalidad	Descripción	Bugs
LIST-SAFE	Navegar como usuario con rol client a la url, comprobar que efectivamente se muestra la vista y no lanza ningún error el sistema.	No se ha encontrado ningún bug.
LIST-HACK	Navegar como usuario sin rol, rol distinto a client o con un client ajeno a la url y comprobar que efectivamente no muestra la vista y que lanza error el sistema.	No se ha encontrado ningún bug.
SHOW-SAFE	Navegar como usuario con rol client y dueño de la entidad, comprobar que efectivamente se muestra la vista y no lanza ningún error el sistema.	No se ha encontrado ningún bug.
SHOW-HACK	Navegar como usuario sin rol client o con un client ajeno a la entidad a la url, comprobar que efectivamente no se muestra la vista y que lanza error el sistema.	No se ha encontrado ningún bug.

<p>CREATE-SAFE</p>	<ul style="list-style-type: none"> - Navegar como usuario con rol client y dueño de la entidad, comprobar que efectivamente se muestra la vista y no lanza ningún error el sistema. - Probar todos los inputs del formulario de acuerdo con la metodología dada en clase: <p>RECORD ID:</p> <ul style="list-style-type: none"> Probar sin record id Probar record id duplicado en la bddd Probar sin letras Probar record id sin las letras del comienzo Probar record id sin la letra del centro Probar sin guiones Probar sin número Probar con menos números Probar con más números <p>COMPLETENESS:</p> <ul style="list-style-type: none"> Probar sin completeness Probar completeness negativo Probar con 0 Probar con 0.99 Probar con 0.01 Probar con 1.01 <p>COMMENT:</p>	<p>No se ha encontrado ningún bug.</p>
--------------------	---	--

	<p> Probar vacío Probar código html Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter </p> <p> RESPONSIBLE PERSON: Probar vacío Probar código html Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter </p> <p> CONTRACT: Probar vacío Probar con un contract del desplegable </p>	
--	---	--

UPDATE-SAFE	<ul style="list-style-type: none"> - Navegar como usuario con rol client y dueño de la entidad, comprobar que efectivamente se muestra la vista y no lanza ningún error el sistema. - Probar todos los inputs del formulario de acuerdo con la metodología dada en clase: <p>RECORD ID:</p> <ul style="list-style-type: none"> Probar sin record id Probar record id duplicado en la bddd Probar sin letras Probar record id sin las letras del comienzo Probar record id sin la letra del centro Probar sin guiones Probar sin número Probar con menos números Probar con más números Probar con el antiguo record id <p>COMPLETENESS:</p> <ul style="list-style-type: none"> Probar sin completeness Probar completeness negativo Probar con 0 Probar con 0.99 Probar con 0.01 Probar con 1.01 	No se ha encontrado ningún bug.
-------------	--	---------------------------------

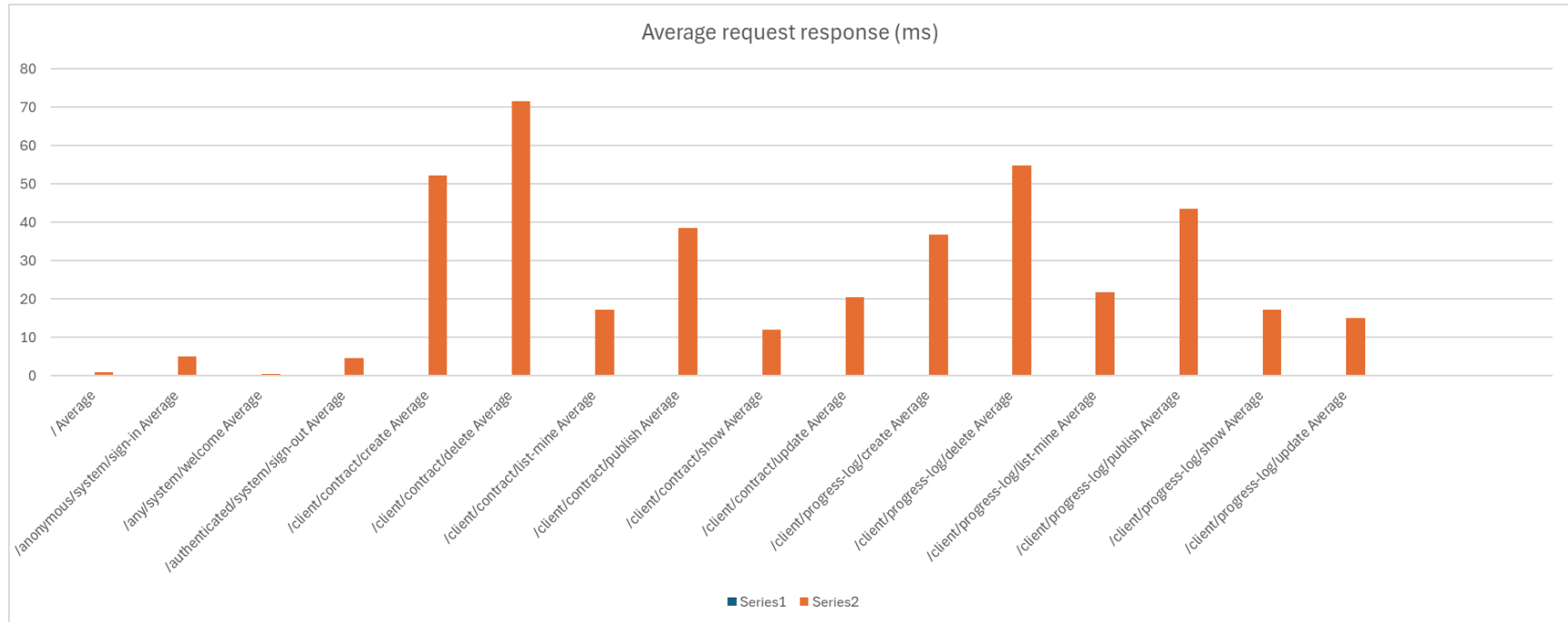
	<p>COMMENT:</p> <ul style="list-style-type: none"> Probar vacío Probar código html Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter <p>RESPONSIBLE PERSON:</p> <ul style="list-style-type: none"> Probar vacío Probar código html Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx Probar con 1 carácter <p>CONTRACT:</p> <ul style="list-style-type: none"> Probar vacío Probar con un contract del desplegable 	
PUBLISH-SAFE	<ul style="list-style-type: none"> - Navegar como usuario con rol client y dueño de la entidad, comprobar que efectivamente se muestra la vista y no lanza ningún error el sistema. - Probar todos los inputs del formulario de acuerdo con la metodología dada en clase: 	

	<p>RECORD ID:</p> <ul style="list-style-type: none"> Probar sin record id Probar record id duplicado en la bdd Probar sin letras Probar record id sin las letras del comienzo Probar record id sin la letra del centro Probar sin guiones Probar sin número Probar con menos números Probar con más números Probar con el antiguo código <p>COMPLETENESS:</p> <ul style="list-style-type: none"> Probar sin completeness Probar completeness negativo Probar con 0 Probar con 0.99 Probar con un decimal pequeño Probar con 1.01 <p>COMMENT:</p> <ul style="list-style-type: none"> Probar vacío Probar código html Probar código de lógica Probar con otro charset Probar con el máx+1 Probar con el máx 	
--	---	--

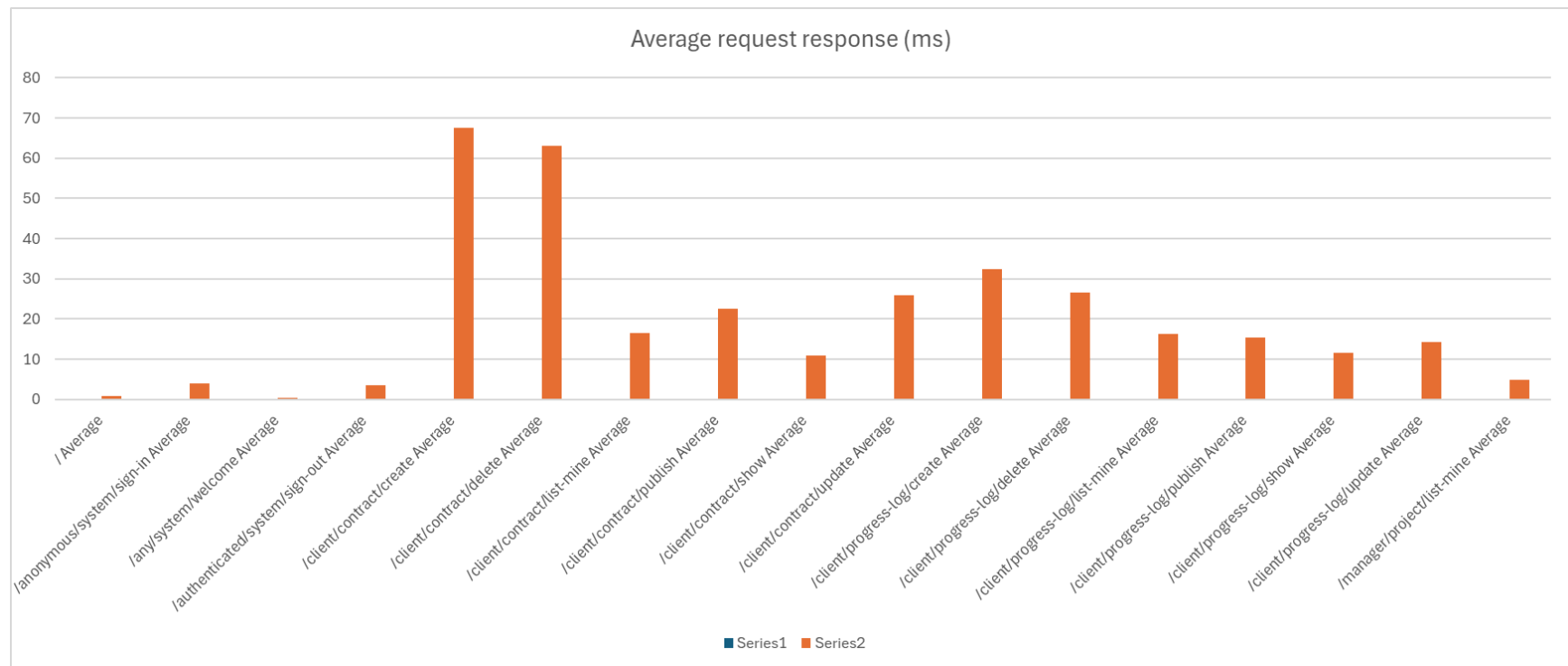
	<p>Probar con 1 carácter</p> <p>RESPONSIBLE PERSON:</p> <p>Probar vacío</p> <p>Probar código html</p> <p>Probar código de lógica</p> <p>Probar con otro charset</p> <p>Probar con el máx+1</p> <p>Probar con el máx</p> <p>Probar con 1 carácter</p> <p>CONTRACT:</p> <p>Probar vacío</p> <p>Probar con un contract del desplegable</p>	
DELETE-SAFE		No se ha encontrado ningún bug.

Testing de rendimiento

A continuación la siguiente tabla muestra los tiempos de respuesta de cada petición del sistema por funcionalidad antes de la optimización del sistema cuyos cálculos se pueden ver más a detalle en los reports de la carpeta de documentación del Student #2 en el entregable D04:



Vemos que los tiempos de peticiones entran en el intervalo de aceptación del cliente.
A continuación la misma tabla tras haber incluido índices en las tablas de la base de datos:



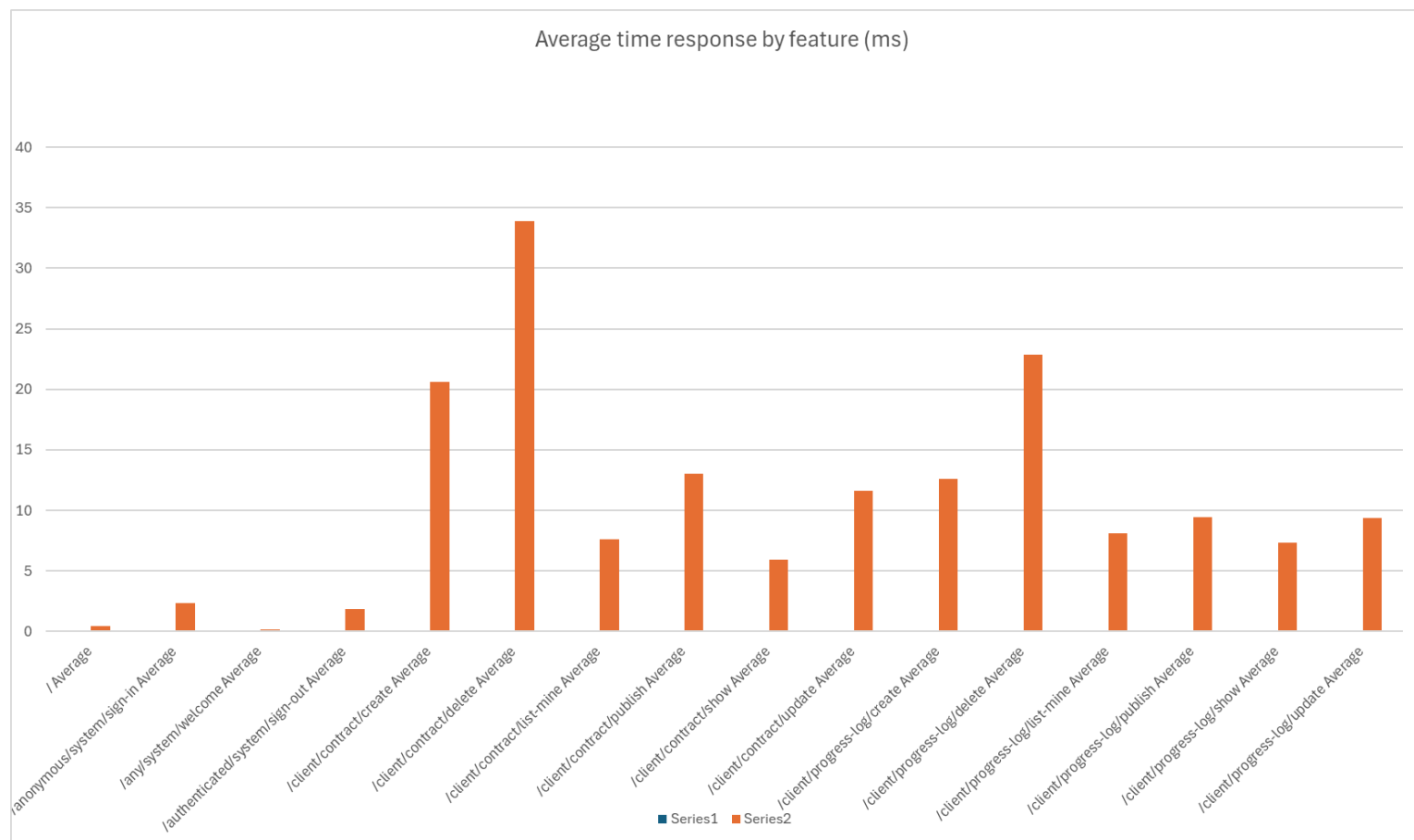
Vemos que los tiempos de peticiones entran en el intervalo de aceptación del cliente y se puede apreciar comparando una mejora general en los tiempos. Veremos el contraste con confianza 95%:

z-Test: Two Sample for Means		
	<i>No annotation</i>	<i>Table annotation</i>
Mean	12.49000763	10.87322164
Known Variance	526.2094414	651.4962313
Observations	552	554
Hypothesized Mean Difference	0	
z	1.107994966	
P(Z<=z) one-tail	0.133931995	
z Critical one-tail	1.644853627	
P(Z<=z) two-tail	0.26786399	
z Critical two-tail	1.959963985	

Como se puede observar de los datos sacados del documento el z-analysis muestra un valor 0.26786, superior al delta de 0.05, por lo que podemos concluir que el sistema no ha mejorado tras la implementación de índices en base de datos pese a haber intentado mejorar el rendimiento con distintas combinaciones de dichos índices.

Finalmente, veremos el z-analysis del mismo sistema, esta vez comparándose con los tiempos en otra máquina.

La siguiente tabla muestra los tiempos de respuesta de cada petición del sistema en otro ordenador usando la versión optimizada.



A continuación veremos el contraste con confianza 95% entre las pruebas ejecutadas en nuestro equipo y el PC 2 tras incluir los índices en la base de datos :

z-Test: Two Sample for Means		
	<i>My times</i>	<i>Other PC times</i>
Mean	10.87322164	4.975031804
Known Variance	651.4962313	64.36396989
Observations	554	547
Hypothesized Mean Difference	0	
z	5.185728947	
P(Z<=z) one-tail	1.07586E-07	
z Critical one-tail	1.644853627	
P(Z<=z) two-tail	2.15172E-07	
z Critical two-tail	1.959963985	

Como se puede observar de los datos sacados del documento, el z-analysis muestra un valor muy inferior al delta de 0.05 y es cercano a cero, por lo que podemos concluir que la ejecución del sistema ha mejorado notablemente al ejecutarse en otro equipo. Esto denota una gran superioridad de rendimiento del PC 2 en comparación con nuestro propio equipo.

Conclusiones

En conclusión, en esta entrega hemos testeado a fondo las funcionalidades de nuestra aplicación, donde hemos encontrado y subsanado algunos bugs, y a la vez pudimos analizar el rendimiento de nuestra aplicación y mejorarla gracias a los índices de la base de datos, aunque dicha mejora no sea relevante como nos muestra el z-analysis. Hemos comparado el rendimiento del sistema en dos ordenadores distintos y hemos concluido cuál de ellos es el más potente gracias al z-analysis. Y por último recalcar la importancia de la selección del hardware para la optimización de los sistemas.

Bibliografía

Intencionalmente en blanco.