Plan de pruebas de software

***[MVC Spring Web]***

***Fecha: [05/07/2021]***

# Tabla de contenido

[Historial de Versiones 4](#_TOC_250021)

[Información del Proyecto 4](#_TOC_250020)

[Aprobaciones 4](#_TOC_250019)

[Resumen Ejecutivo 5](#_TOC_250018)

[Alcance de las Pruebas 5](#_TOC_250017)

[Elementos de Pruebas 5](#_TOC_250016)

Nuevas Funcionalidades a Probar ¡Error! Marcador no definido.

Pruebas de Regresión ¡Error! Marcador no definido.

[Funcionalidades a No Probar 14](#_TOC_250015)

Enfoque de Pruebas (Estrategia) ¡Error! Marcador no definido.

[Criterios de Aceptación o Rechazo 14](#_TOC_250014)

[Criterios de Aceptación o Rechazo 14](#_TOC_250013)

[Criterios de Suspensión 14](#_TOC_250012)

[Criterios de Reanudación 15](#_TOC_250011)

[Entregables 15](#_TOC_250010)

[Recursos 15](#_TOC_250009)

[Requerimientos de Entornos – Hardware 15](#_TOC_250008)

[Requerimientos de Entornos – Software 15](#_TOC_250007)

[Herramientas de Pruebas Requeridas 16](#_TOC_250006)

[Personal 16](#_TOC_250005)

[Entrenamiento 16](#_TOC_250004)

[Planificación y Organización 17](#_TOC_250003)

[Matriz de Responsabilidades 17](#_TOC_250002)

Cronograma ¡Error! Marcador no definido.

[Dependencias y Riesgos 17](#_TOC_250001)

[Glosario 18](#_TOC_250000)

# Historial de versiones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fecha** | **Versión** | **Autor** | **Organización** | **Descripción** |
| **05/07/2021** | **1.0** | BRAHAM SEBASTIAN MARIN CASTAÑEDA | **SENA** |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Información del proyecto

|  |  |
| --- | --- |
| Empresa / Organización | SENA |
| Proyecto | CRUD ALQUILAR PATINETAS |
| Fecha de preparación |  |
| Cliente | SENA-OSCAR IVAN BENAVIDEZ |
| Patrocinador principal | BRAHAM SEBASTIAN MARIN CASTAÑEDA |
| Gerente / Líder de proyecto | BRAHAM SEBASTIAN MARIN CASTAÑEDA |
| Gerente / Líder de pruebas de software | BRAHAM SEBASTIAN MARIN CASTAÑEDA |

# Aprobaciones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nombre y Apellido** | **Cargo** | **Departamento u**  **organización** | **Fecha** | **Firma** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Resumen ejecutivo

Este plan de pruebas de software para el proyecto web ALQUILAR-PATINETAS-CRUD hecho en java versión 8.0, donde el propósito de este proyecto es aprender, evidenciar e implementar herramientas para crear páginas web con el patrón arquitectónico MVC usando el framework de java llamado spring MVC donde implementamos las librerías de JSTL que son etiquetas nativas de java que se pueden usar tipo HTML, también se da uso a la librería de JDBC témplate que se encarga de gestionar lo que es la conexión y en si la parte de consultas de bases de datos y a su vez las herramientas jMeter para hacer pruebas de stress y de carga a una aplicación.

El alcance de este proyecto es el aprendizaje profesional como lo implementan hoy día en las compañías el uso de pruebas de stress y de carga , MVC, donde este proyecto local se le realizaron pruebas hasta donde nos dejó el presupuesto ya que el proyecto no se encuentra alojado en un hosting como tal , esto para poder evidenciar el peligro y vulnerabilidad de página web al hacerle una carga de usuarios en un tiempo determinado donde solicitan peticiones al mismo tiempo y donde el reporte generado nos mostrara si los recursos soportan la carga de dichos usuarios y así mismo hacer el estudio correspondiente para determinar si se necesita más presupuesto para darle mayor potencia a la aplicación.

Por otro lado, no se pudo hacer una prueba con un máximo de usuarios (500 usuarios en adelante) ya que los recursos con los que cuenta el funcionario que hace las pruebas no son suficientes para dar un máximo procedimiento para mirar en que momento el programa empieza a tener fallas ya que esta colapsa

# Alcance de las pruebas

## Elementos de pruebas

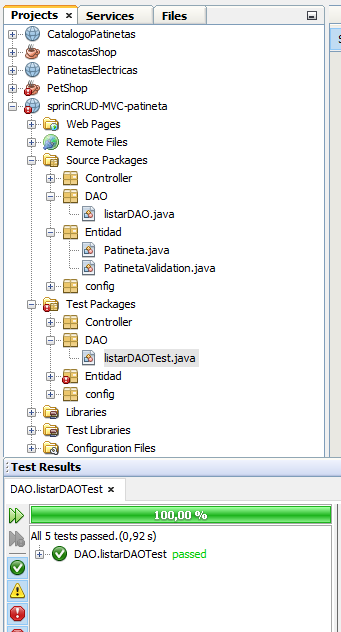
Listado de todos los módulos, componentes o elementos que se van a probar. Si es de alto nivel, se listan las áreas funcionales (módulos o procesos que cubre el

Testing), por otro lado, si es de un nivel detallado se listan los programas, unidades o módulos.

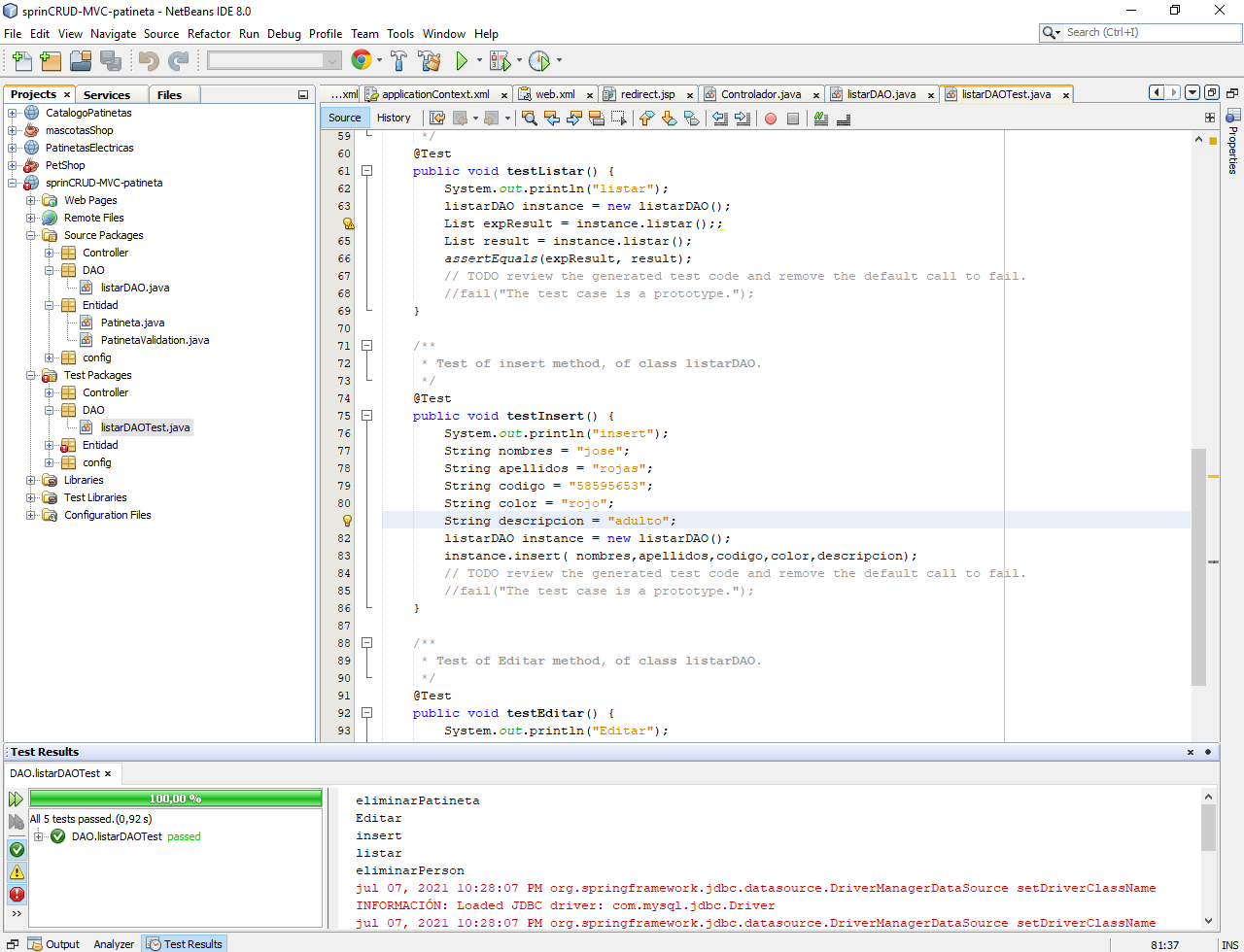
Los módulos de alto nivel a probar con la librería de jUnit y jMeter respectivamente fueron:

JUnit:

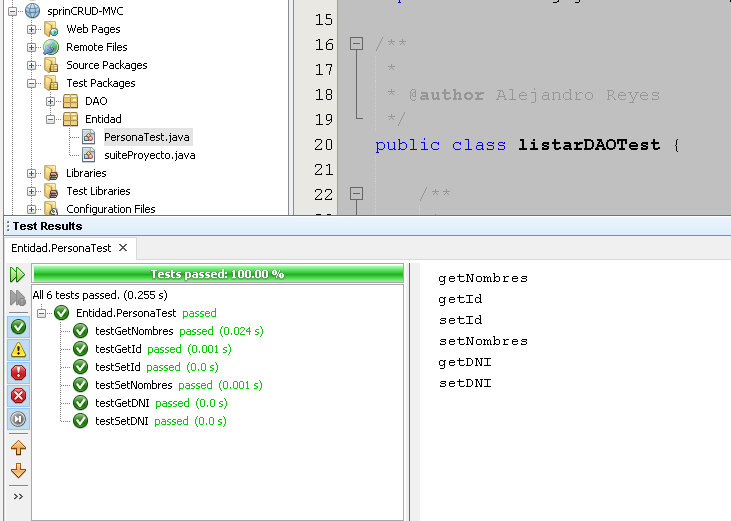
ControllerDAO:



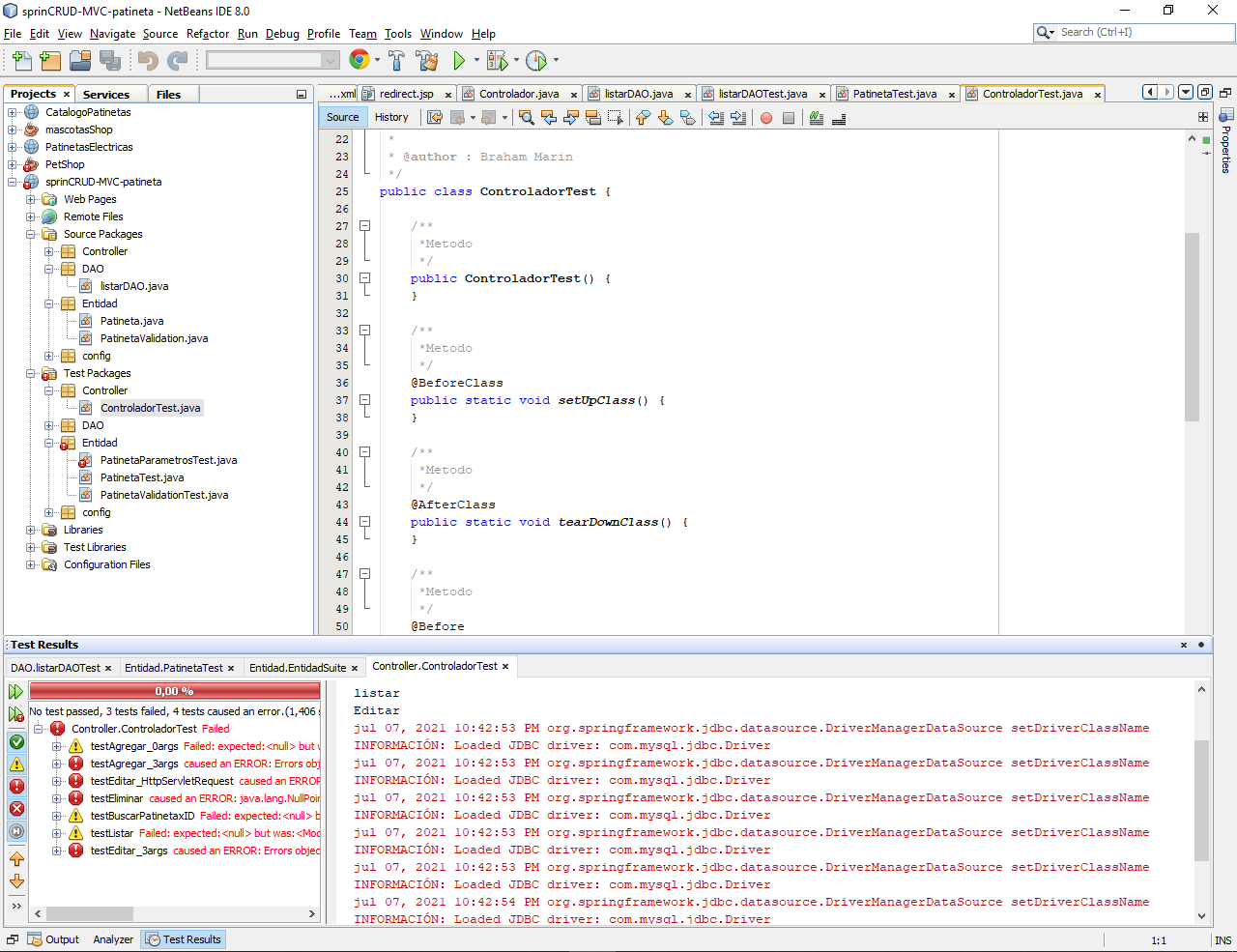
ListarDAO: donde este contiene los métodos el CRUD como lo son (editar, actualizar, eliminar y listar datos)



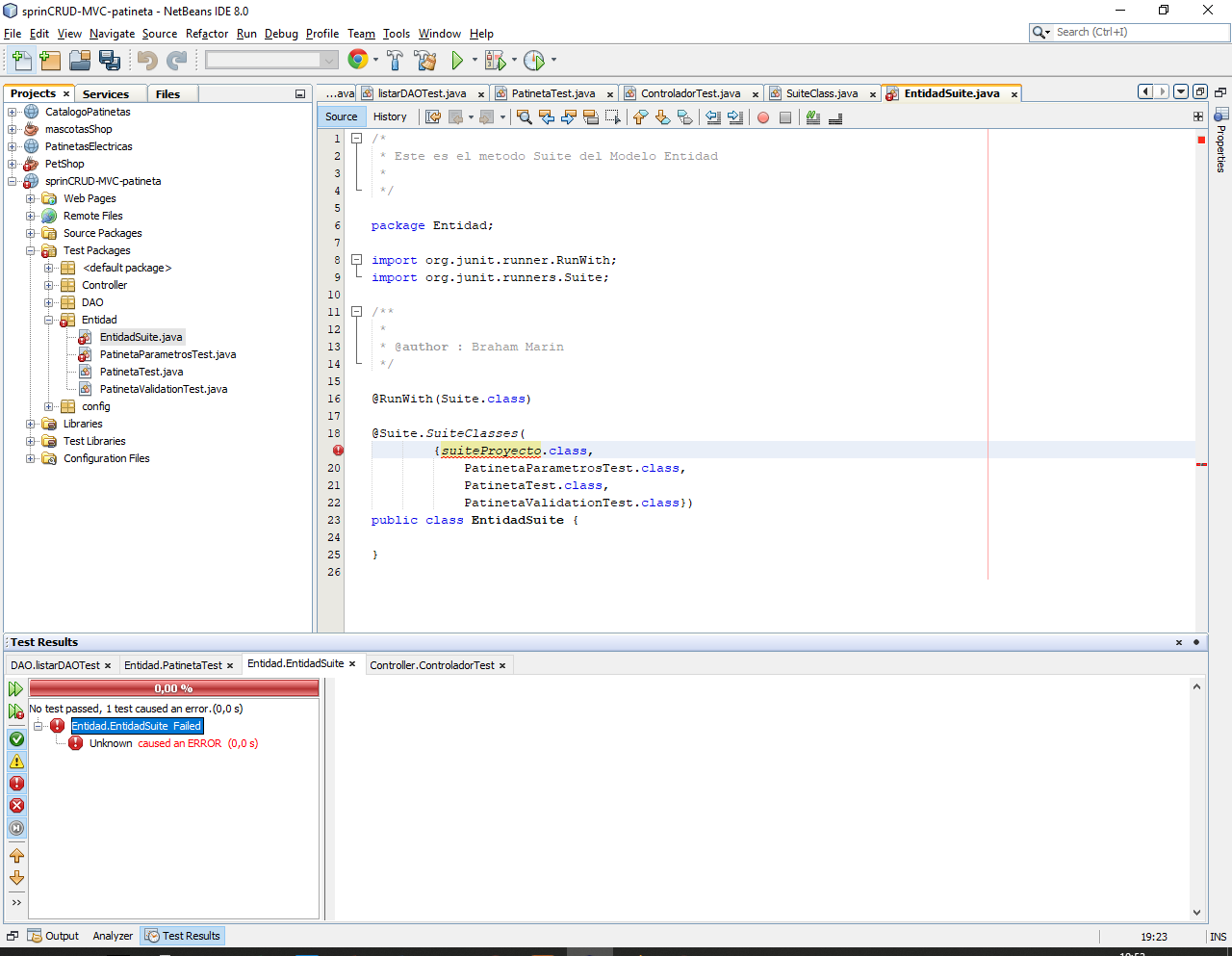
Patineta (MODELO): en este modelo se realizan pruebas para el funcionamiento de los métodos get y set



Este es (Controlador) que se encarga de ralizar la admisla administración de las peticiones.



Suite de clases

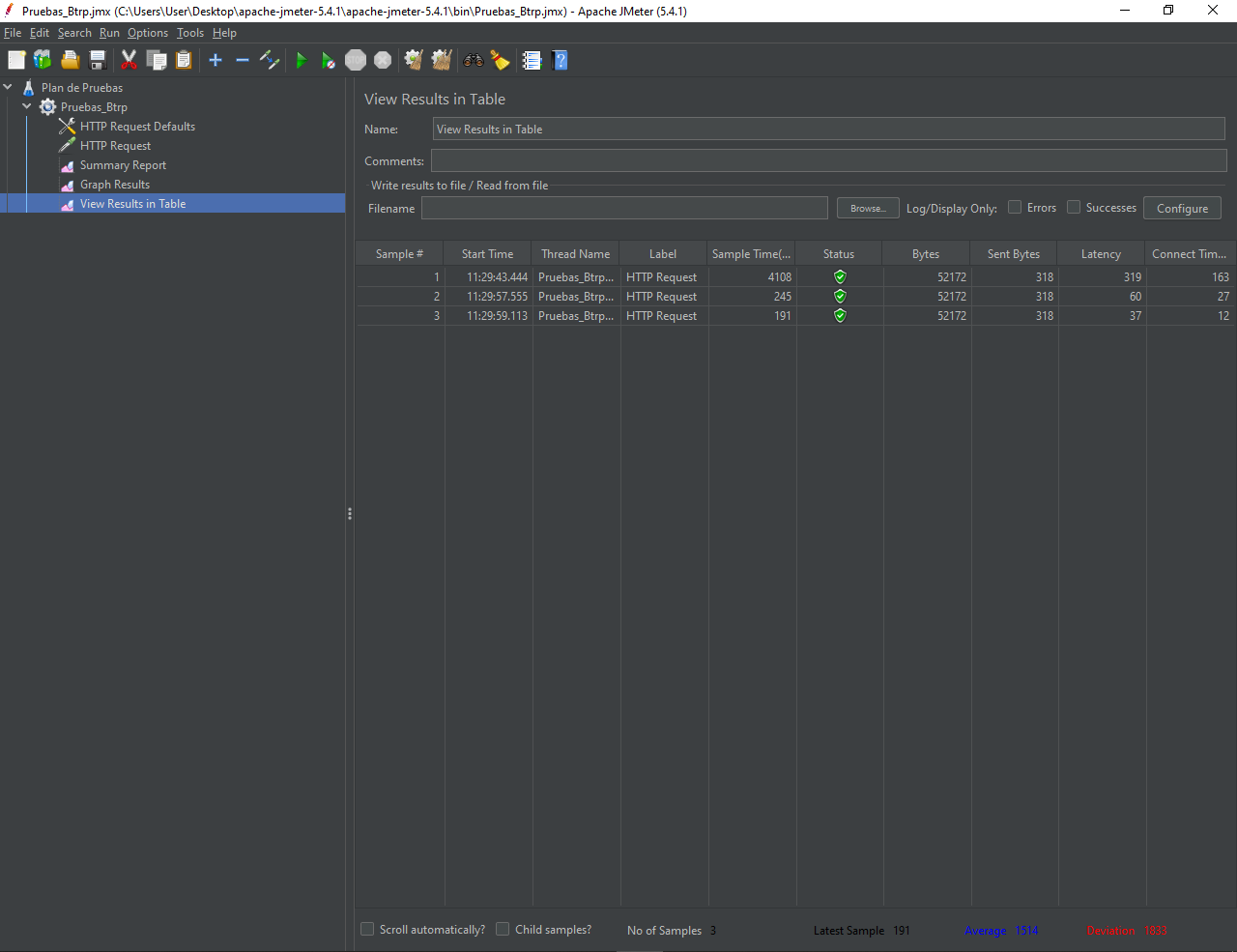


jMeter:

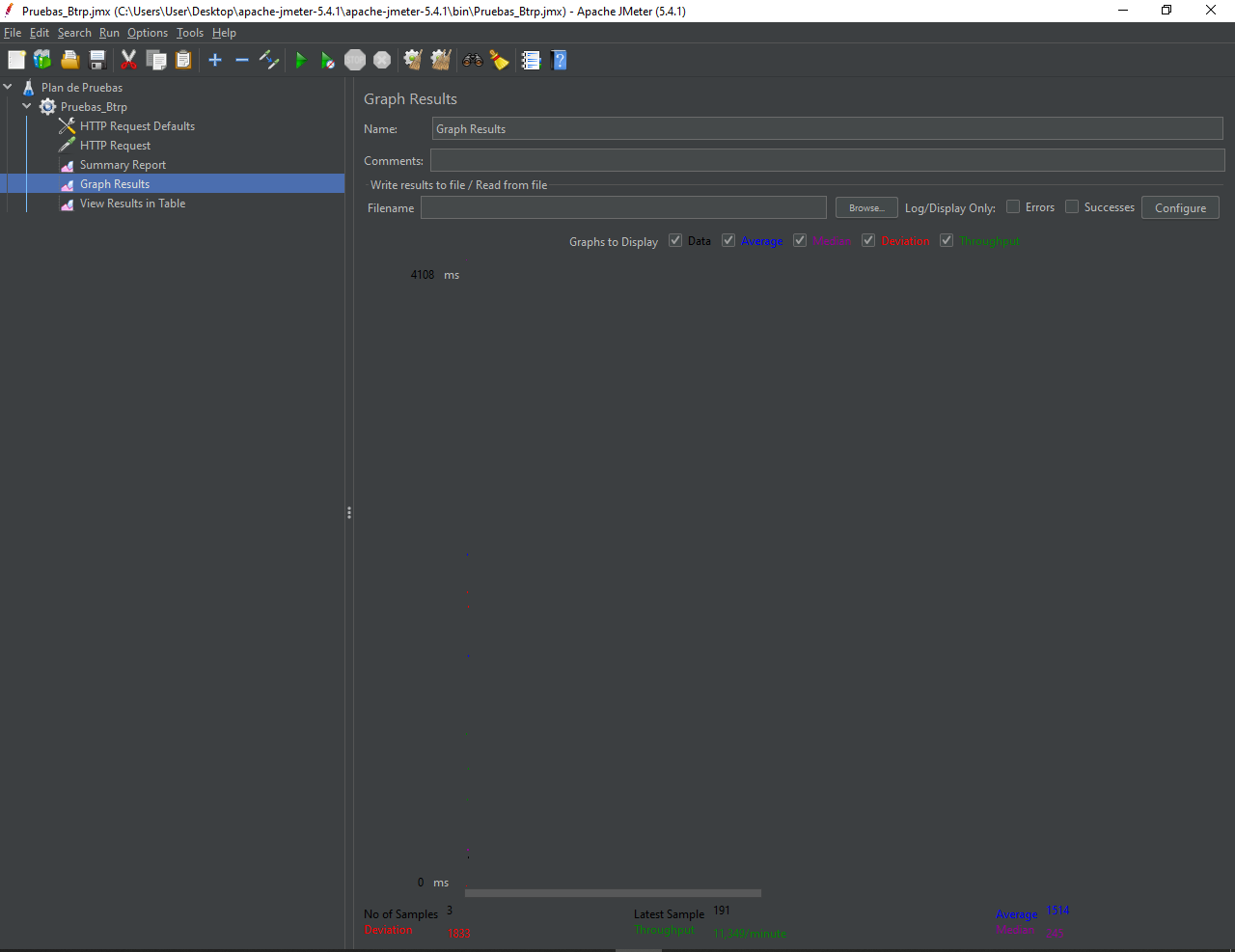
Se probaron al módulo introduccion de aprendizaje de la página de la página de Bootstrap donde las pruebas fueron de stress y de carga y se evaluó con:

1. 1 usuario en 1 segundo

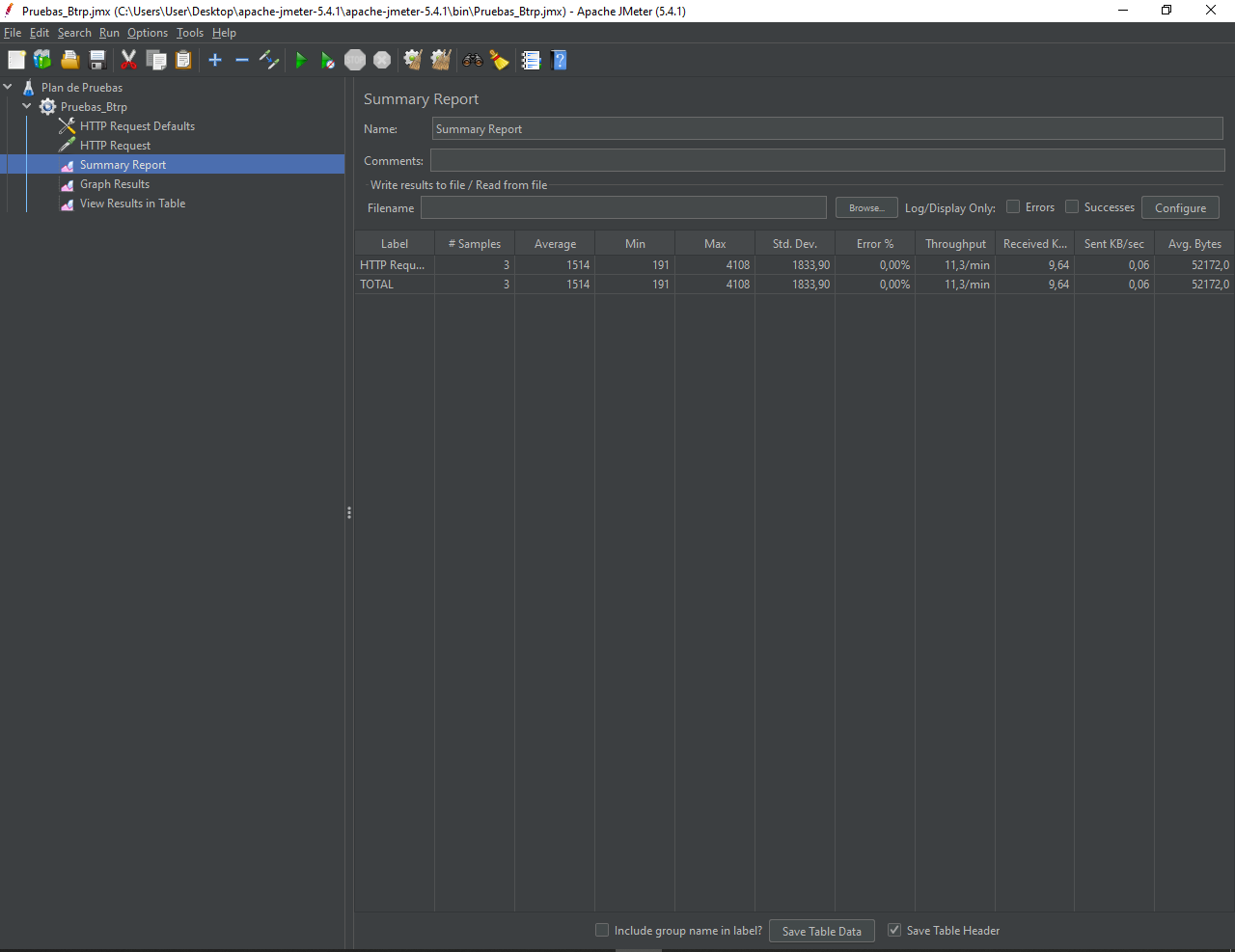
Resultados en tabla



Resultados en Grafico

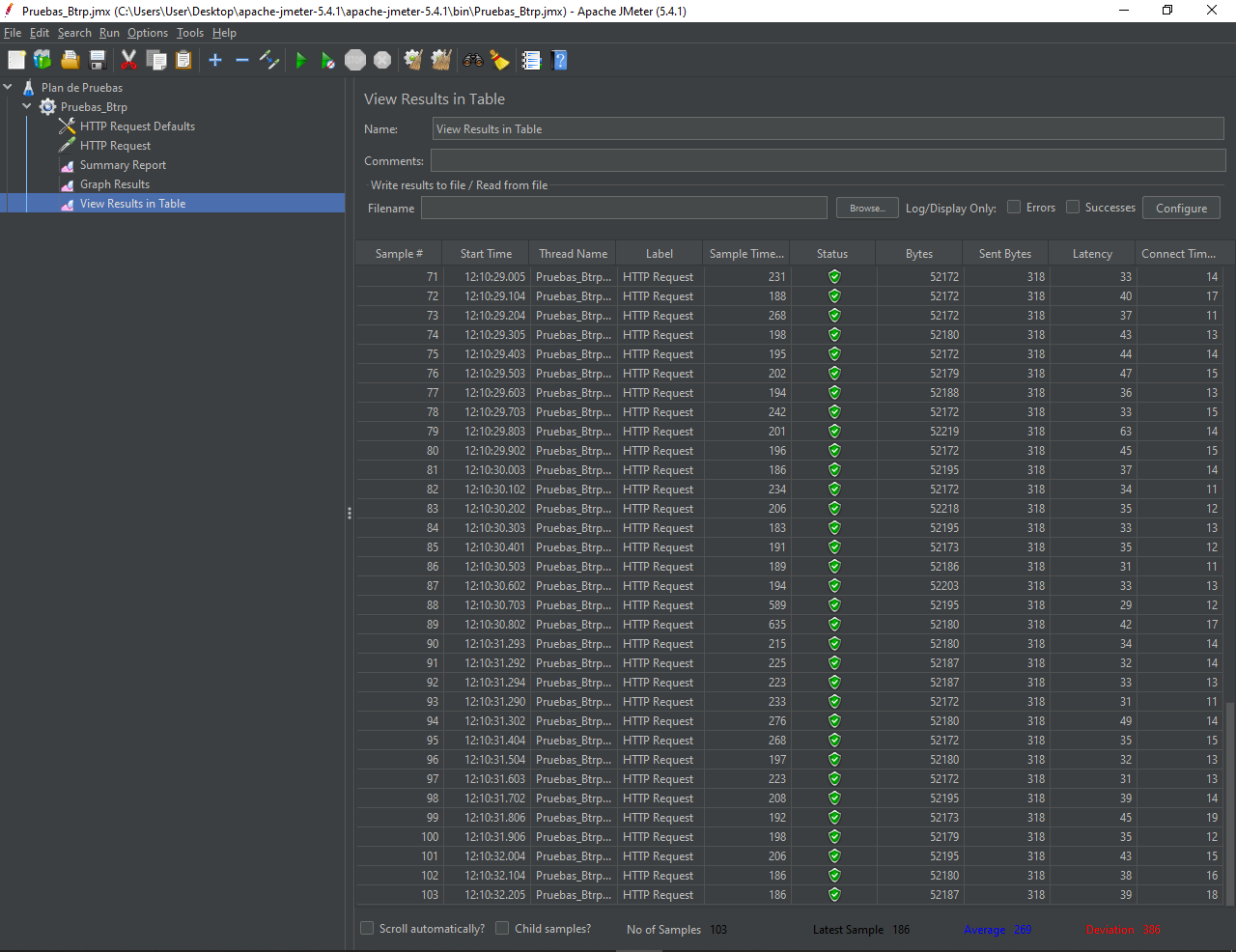


Reporte sumario

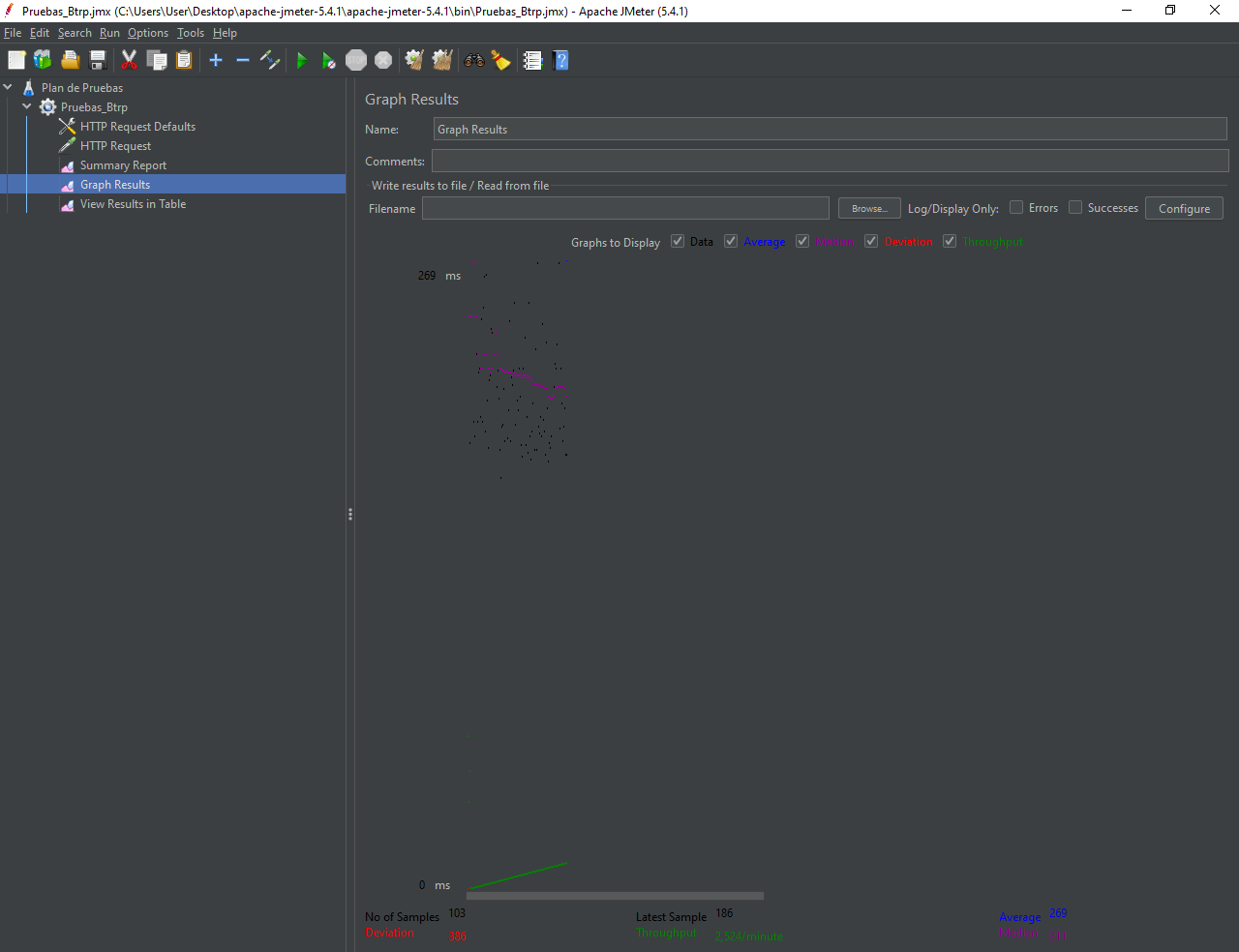


1. 100 usuarios en 10 segundos

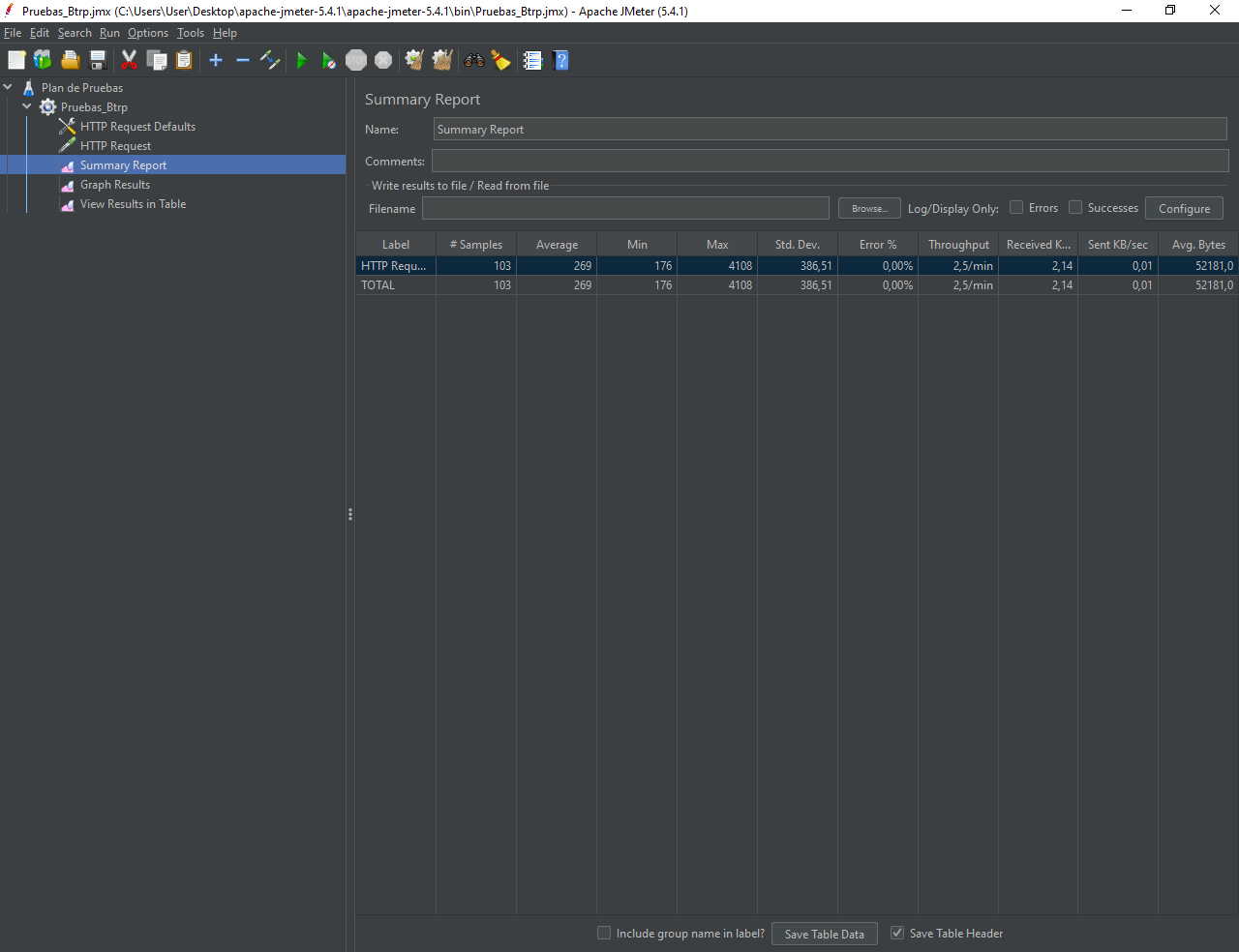
Resultados en tabla



Resultados en grafico

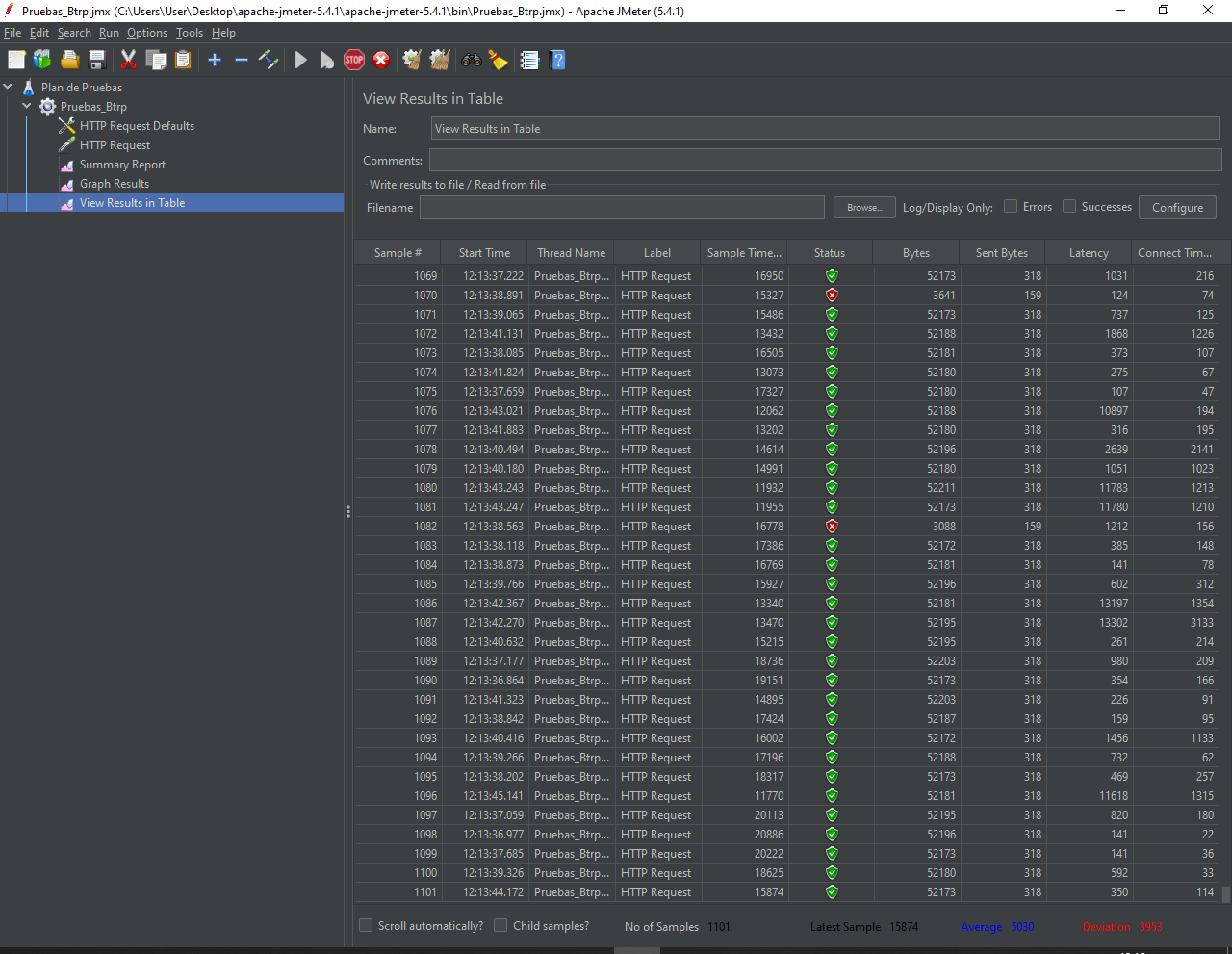


Sumario de reportes

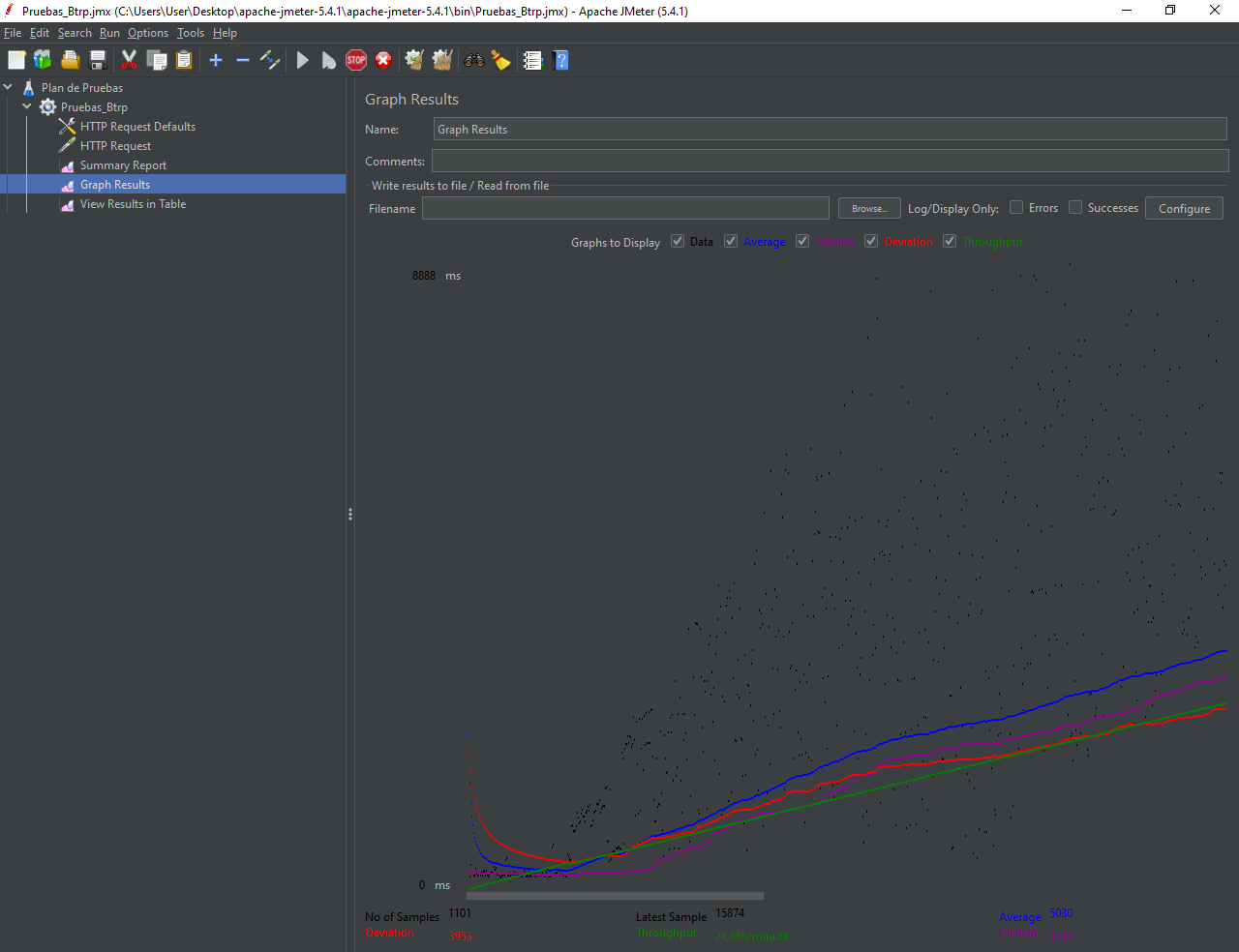


1. 1000 usuarios en 10 segundos Aquí aun se conserva la estabilidad del servidor

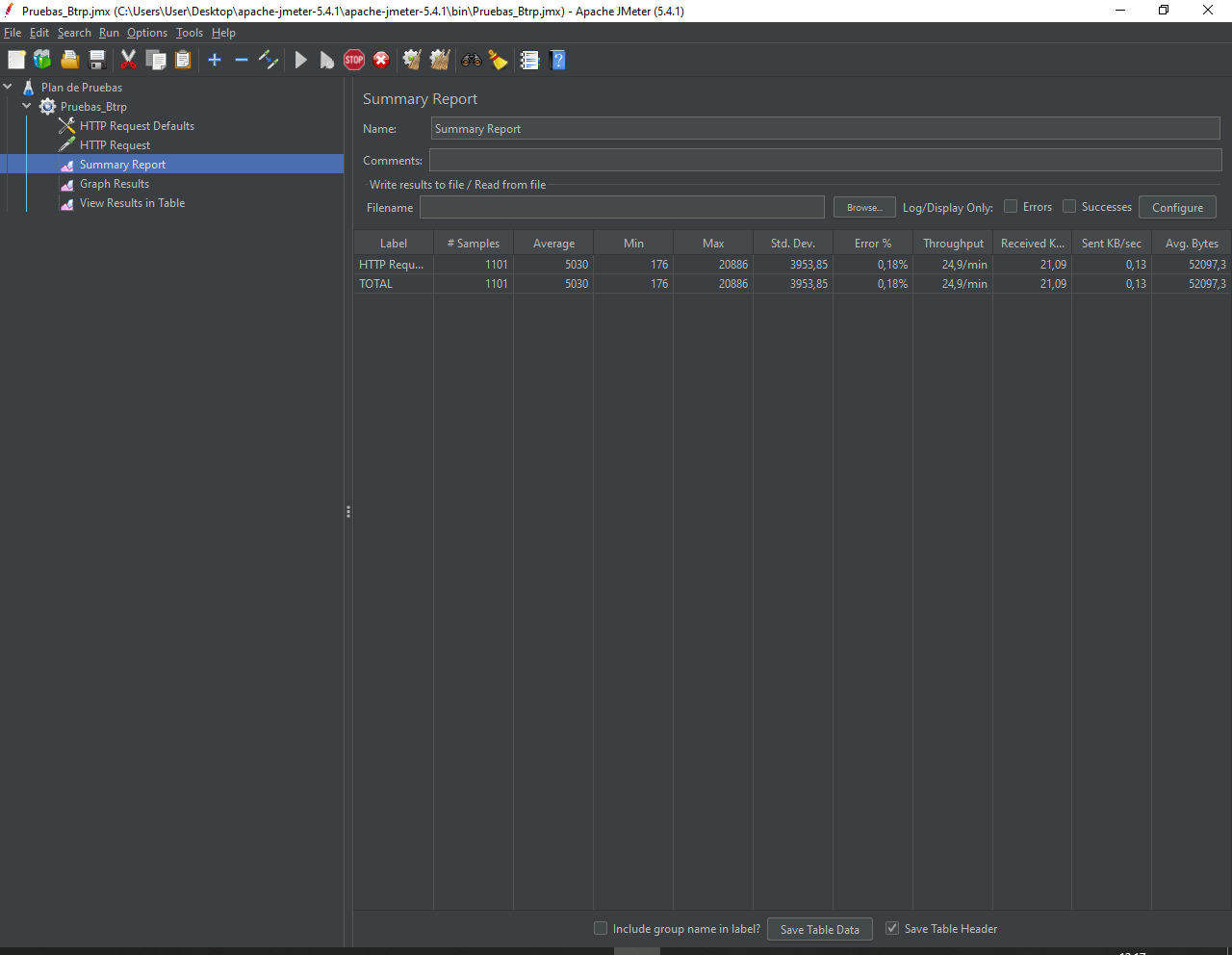
Resultados en tabla



Resultados en grafico

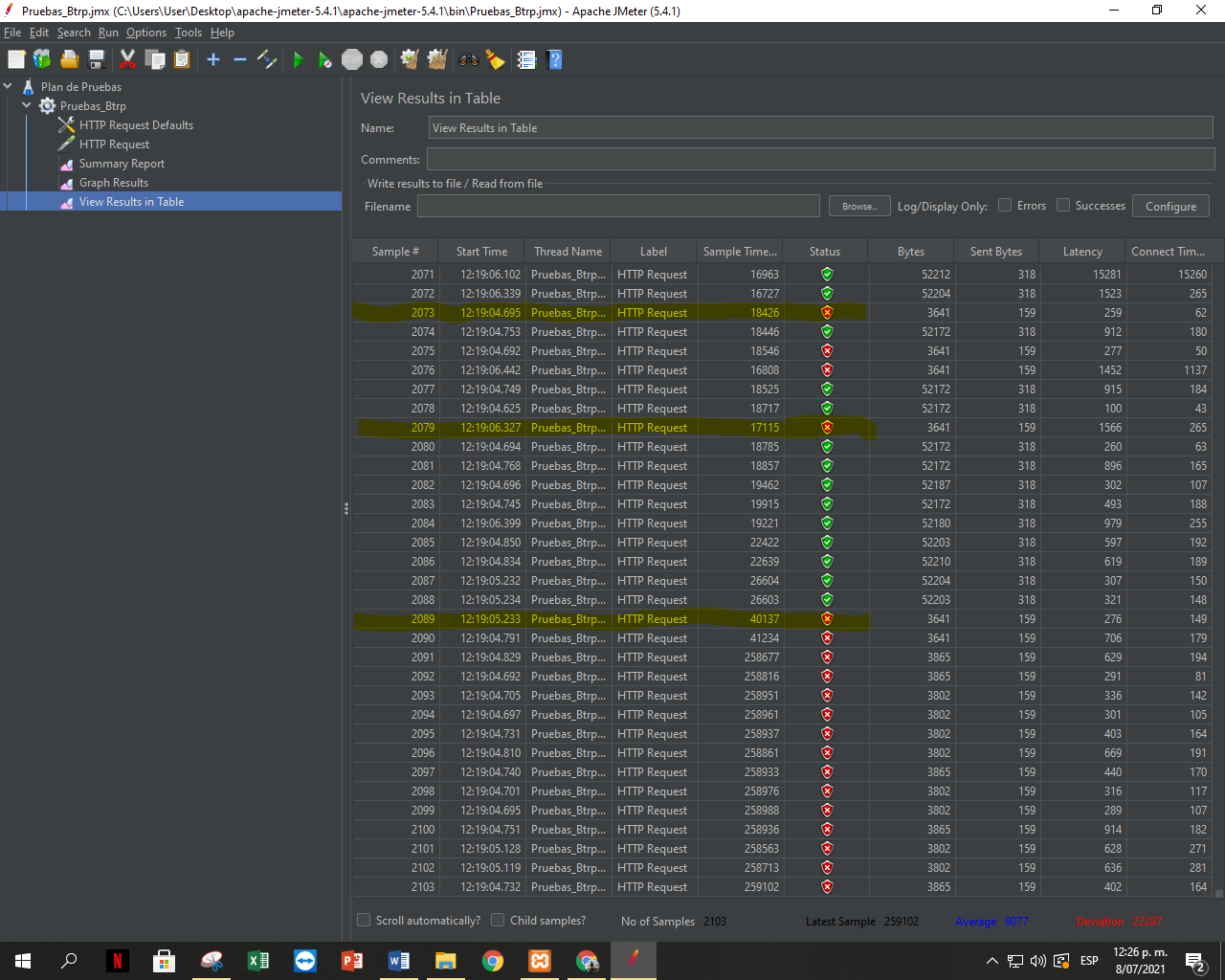


Sumario de reportes

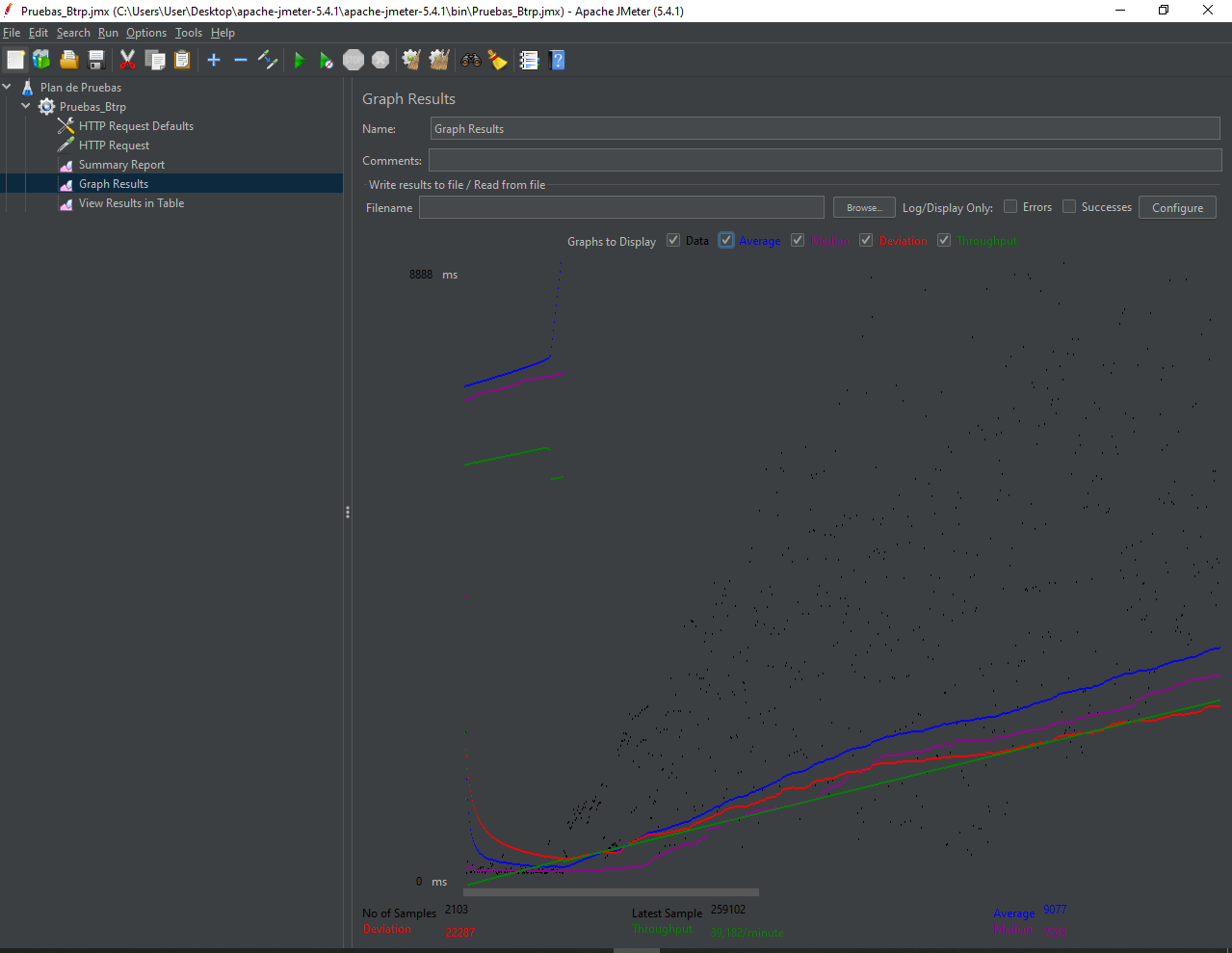


4. 1000 usuarios en 1 segundo. ya con estos parámetros notamos que el servidor comienza a presentar errores

Resultados en tabla



Resultados en grafico



Sumario de reportes

Esto para determinar si la página respondía satisfactoriamente y se evidencia que la pagina tiene una regular recepción de datos y manejo de los mismos cuando hay una carga de 1000 usuarios en 10 segundos, como se muestra en las imágenes.

:

Los informes generados con este método de prueba para la página de Bootstrap y sus módulos de introducción y forms.

1. 1 usuario en 1 segundo

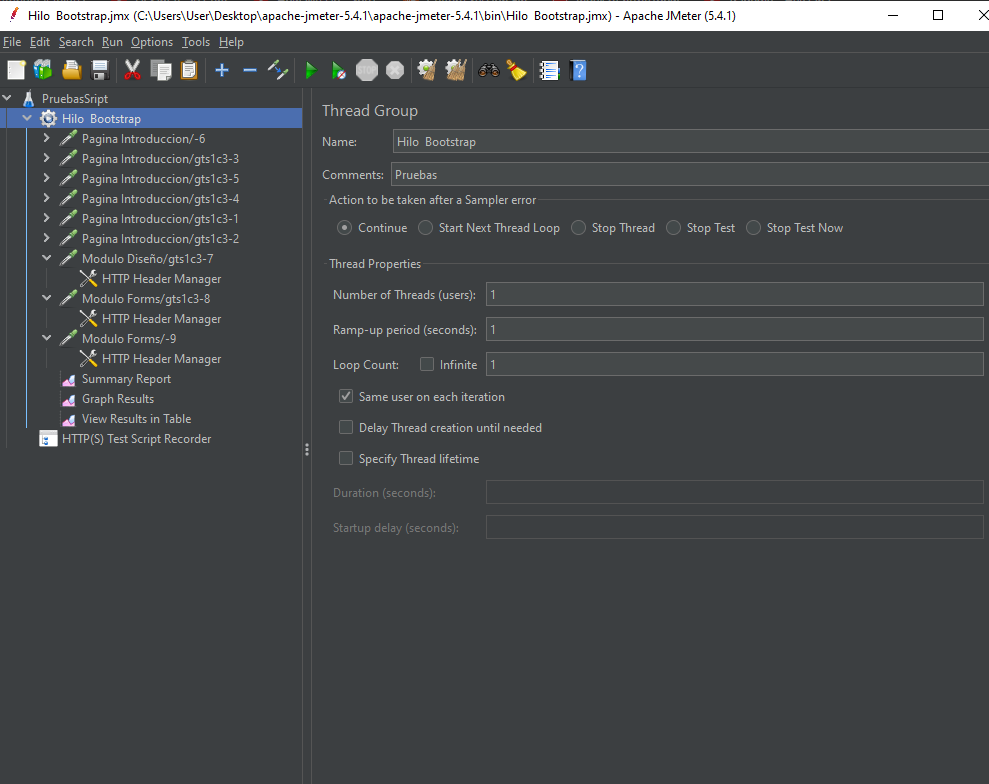
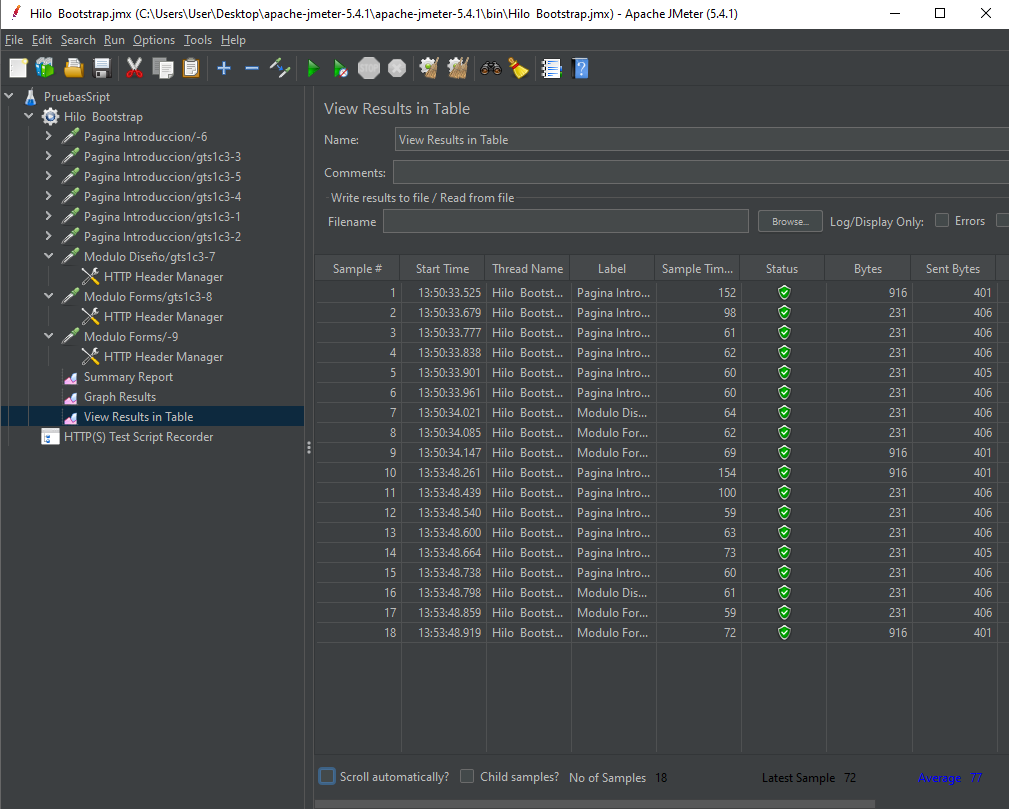
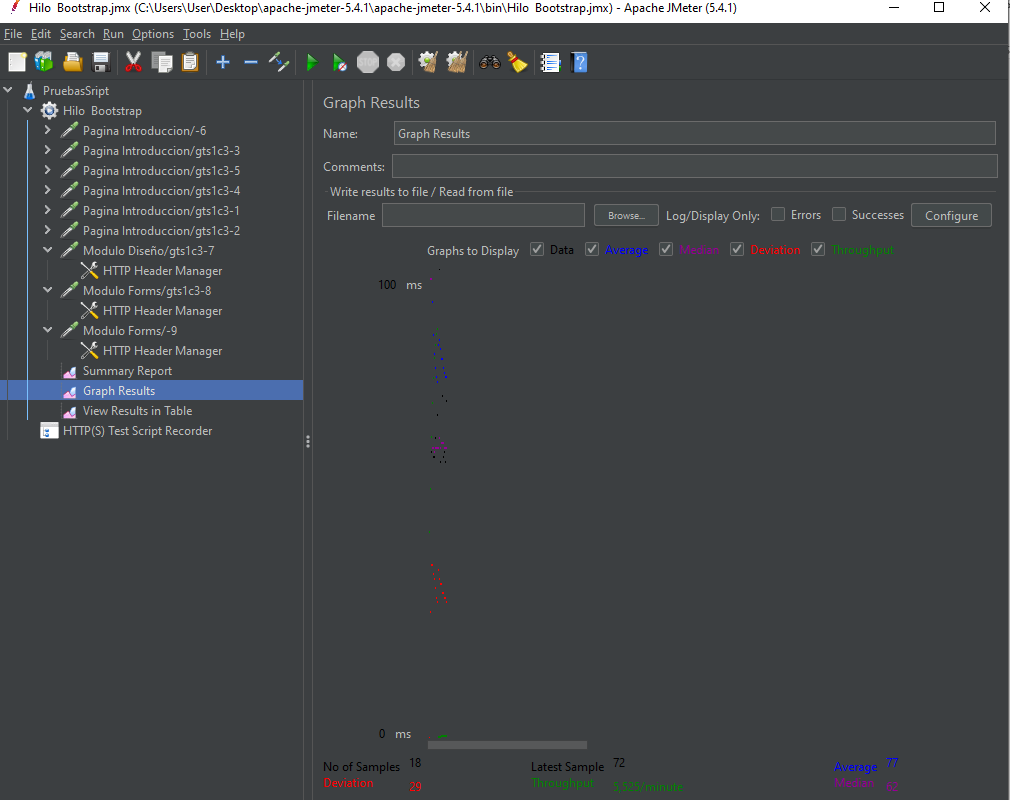


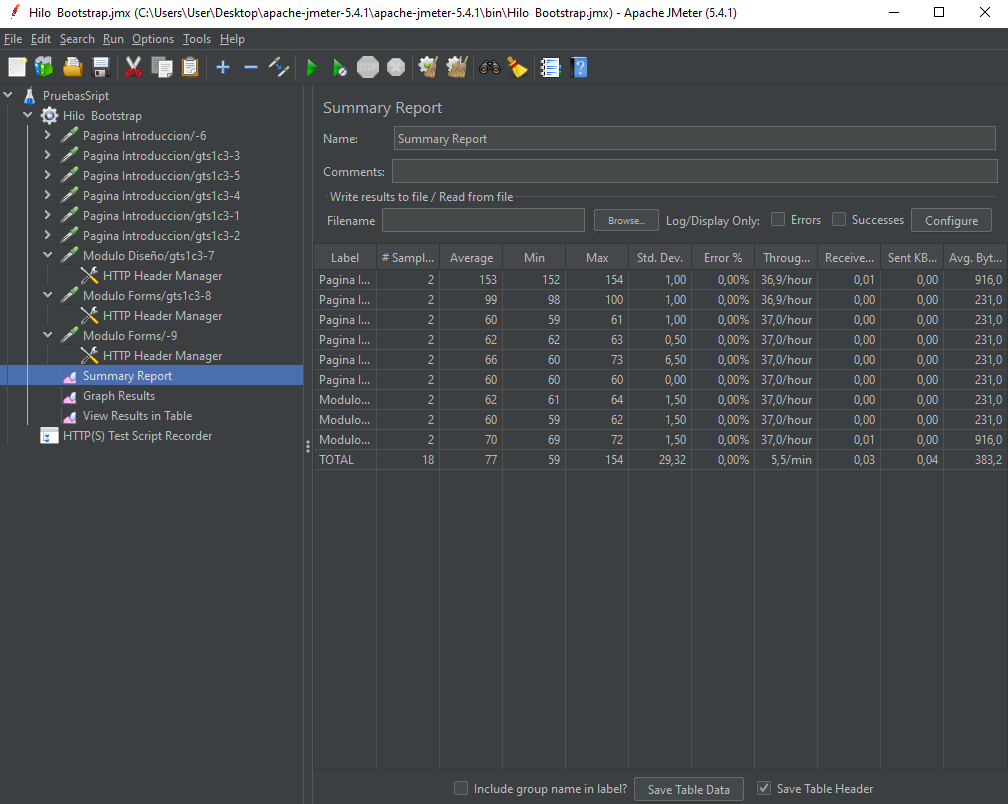
Tabla de resultados



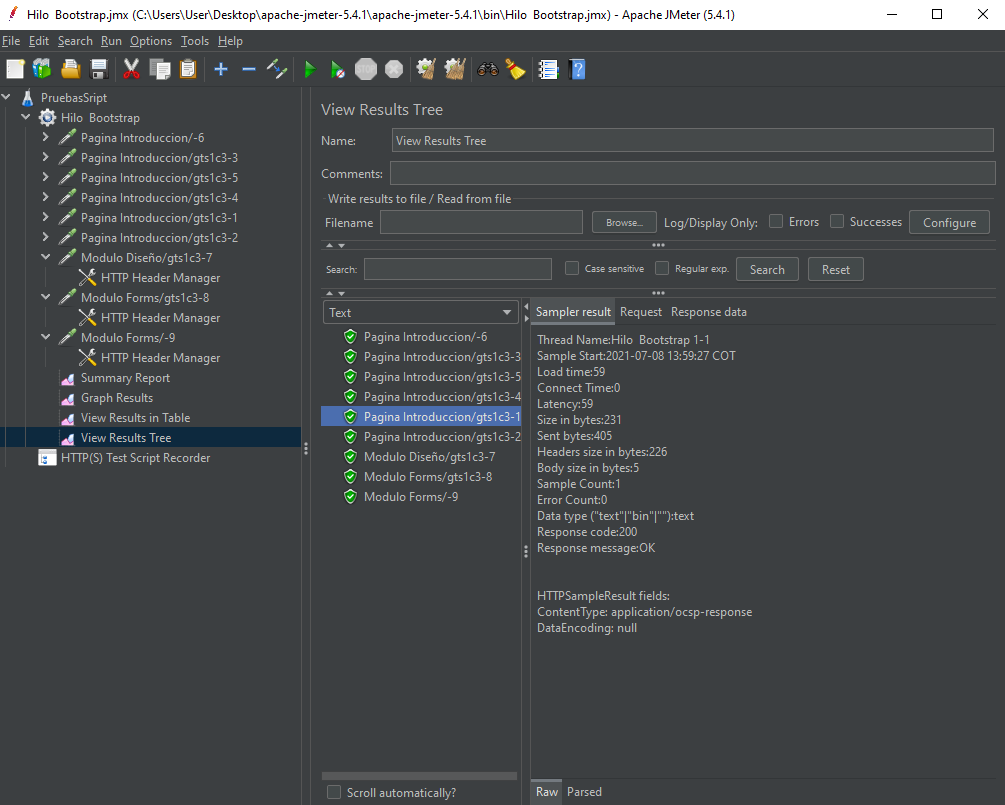
Resultados en Gráficos



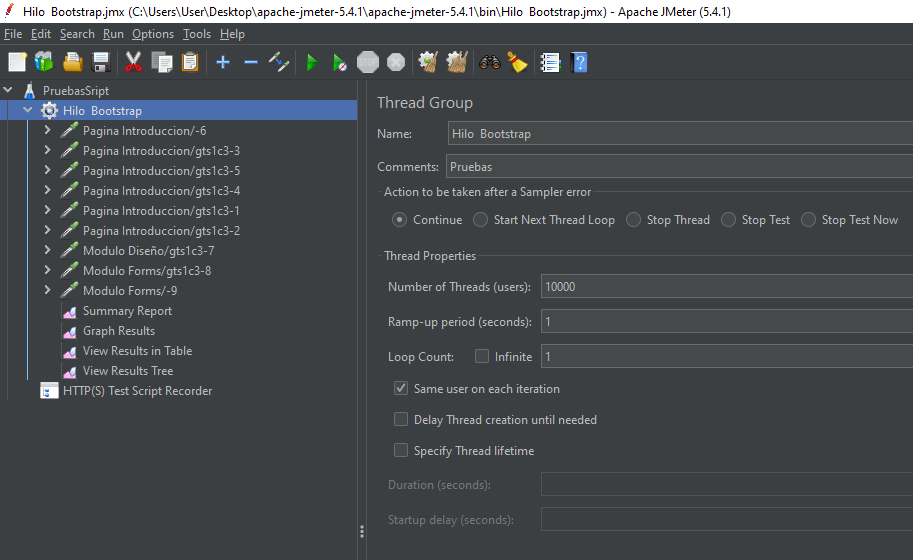
Summary report



Resultados en Arbol

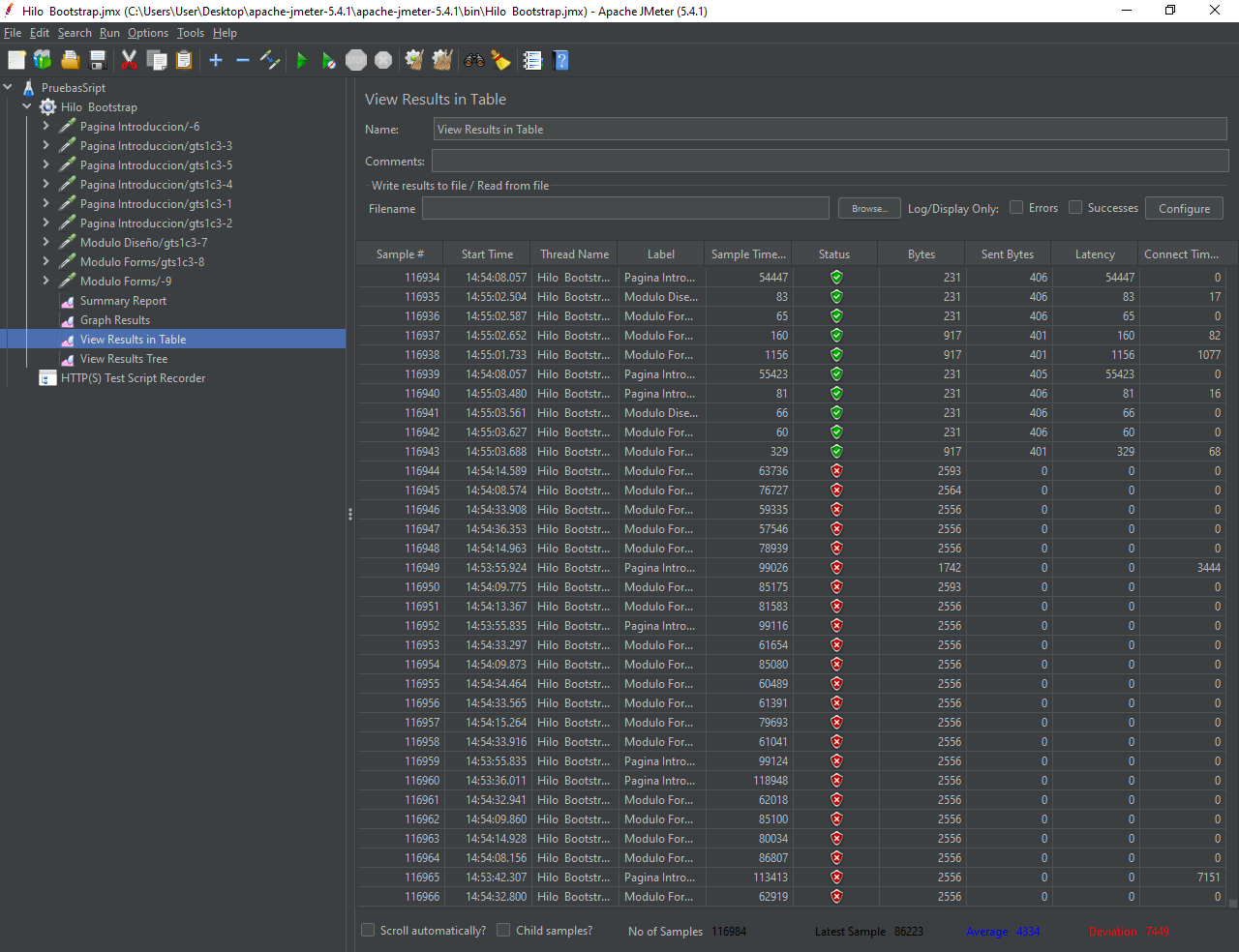


1. 1000 usuarios en 10 segundos

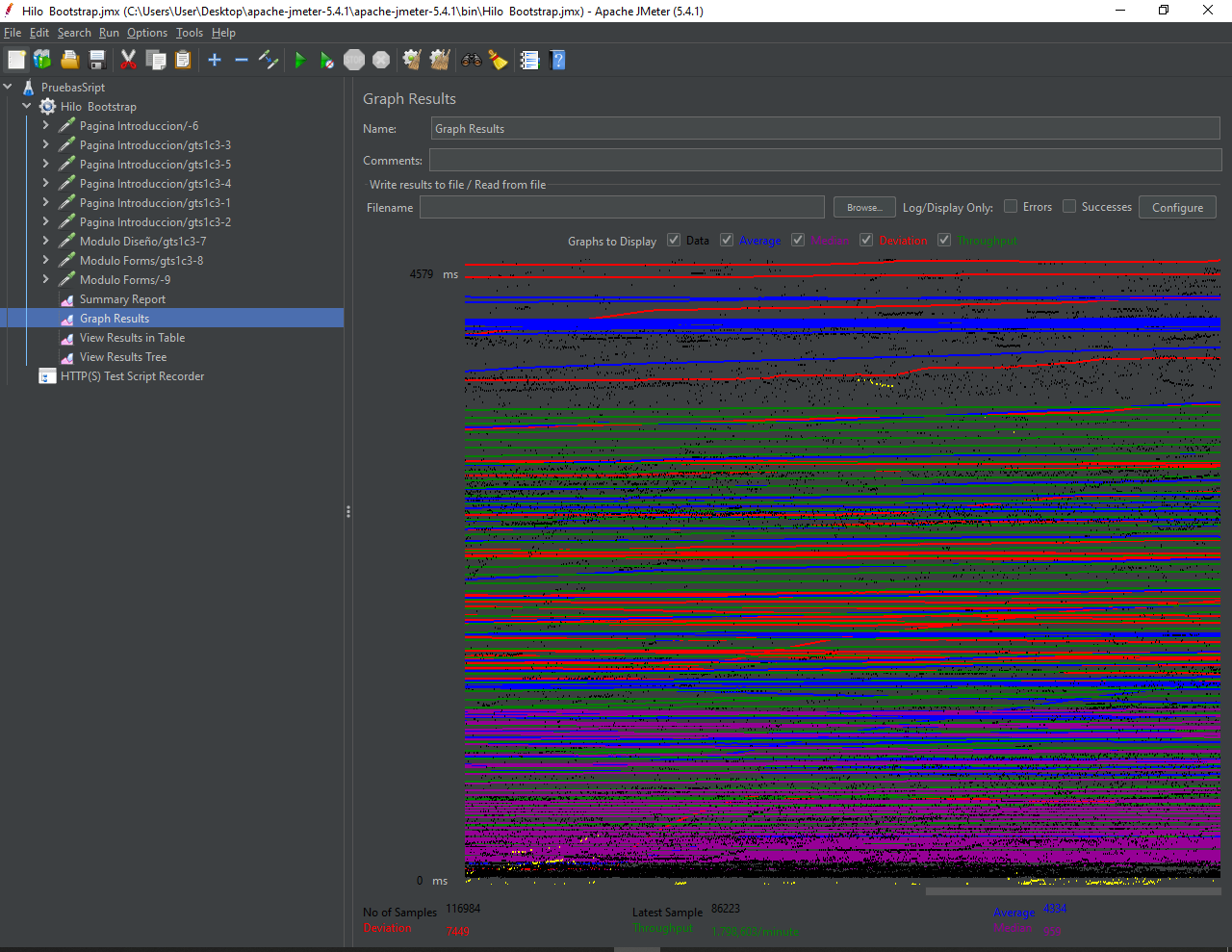


Resultados en tabla:

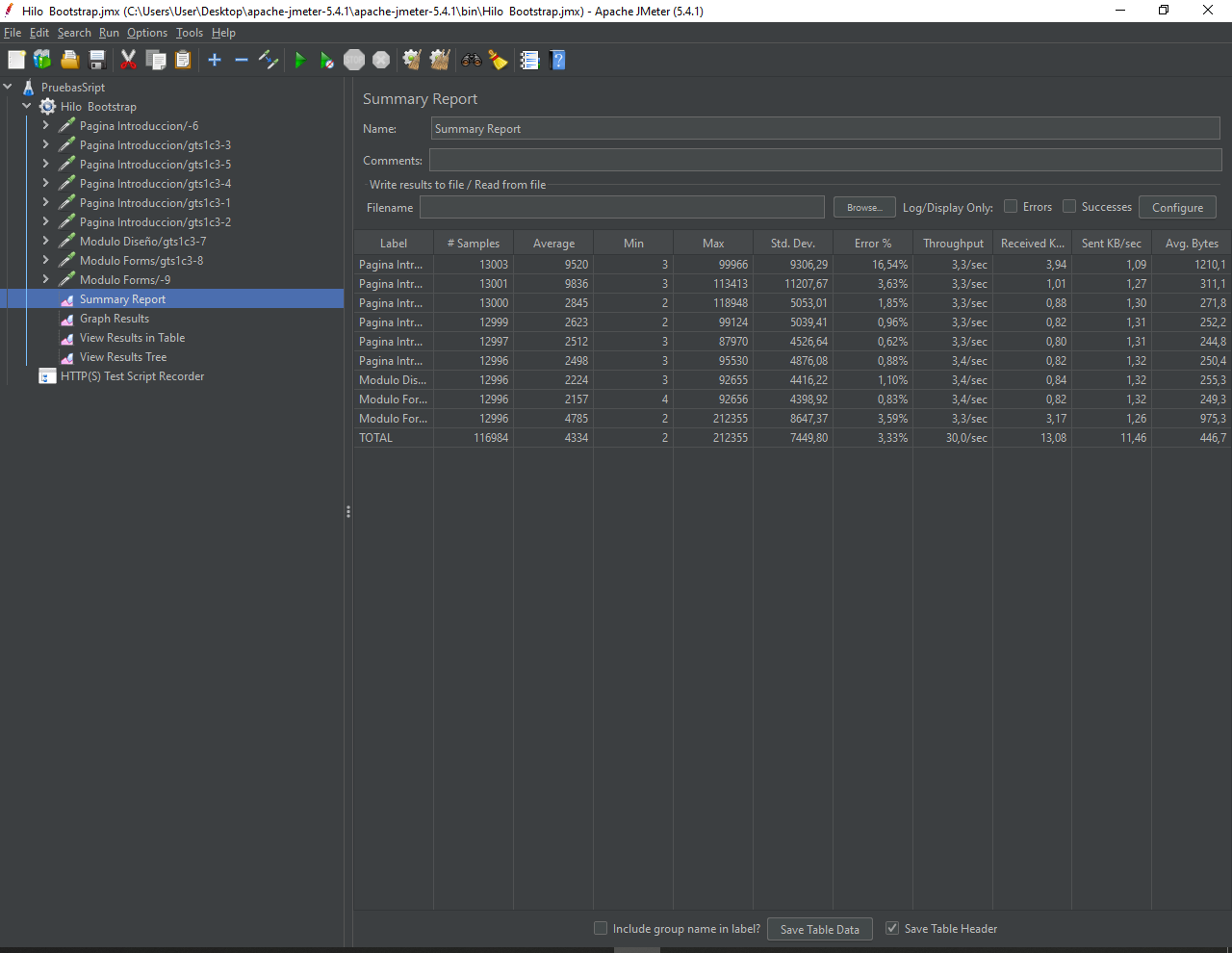
En este proceso con esa carga de usuarios al momento de recibir 116944 usuarios Comienza a presentar error el servicio



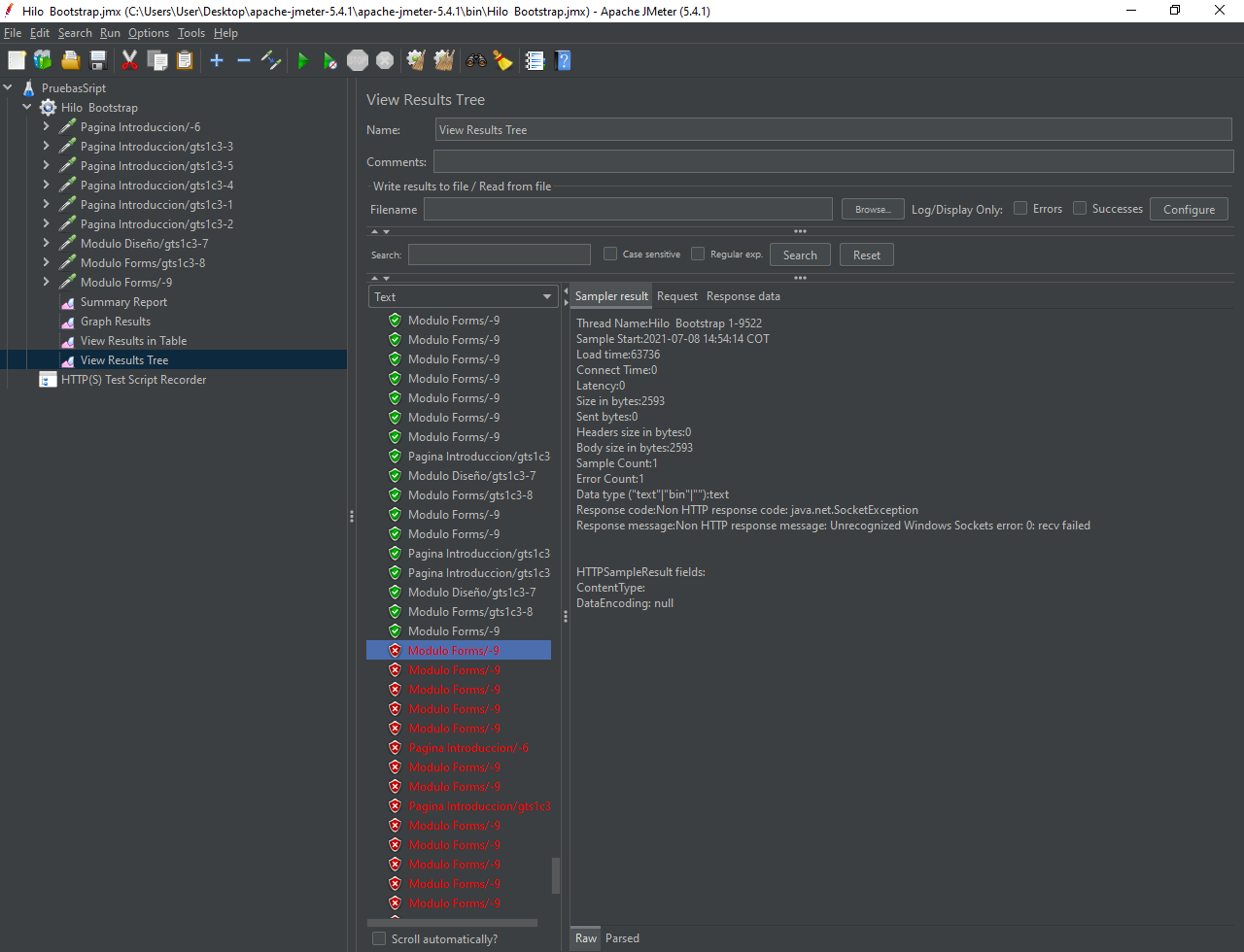
Resultados en grafico



Reporte sumario: En este reporte se visualiza un alto porcentaje de errores pero a medida de procesar la información va disminuyendo la tasa de error



Arbol de reportes



## Funcionalidades a no probar

Las funcionalidades que no se usaran son la validación como tal del ingreso a la plataforma y el nombre usando expresiones regulares, donde el riesgo que se corre es que algún usuario no ingrese su nombre real y/o documento correcto lo que genera una vulnerabilidad del programa ya que solo se evaluó la función sin validar campos vacíos

# Criterios de aceptación o rechazo

## Criterios de aceptación o rechazo

Son los criterios que serán considerados para dar por completado el plan de pruebas de software, por ejemplo: Completar 100% de pruebas unitarias, cierto porcentaje de casos exitosos, cobertura de todos los componentes y líneas de código, porcentaje de defectos corregidos, entre otros.

## Criterios de suspensión

Como es la primera versión se establece que si los métodos que reciben datos llegan a recibir datos nulos se debe suspender temporalmente estos métodos ya que son los principales para el funcionamiento correcto de la aplicación y corregís este aspecto de validación de los datos que se ingresar en estos campos

## Criterios de reanudación

Luego de haber realizado la corrección de los métodos de validación y que su funcionamiento sea el esperado por ejemplo el ingreso a la plataforma se procede habilitar este método, ya que no cumplía con los requerimientos totales, y pues esto generaba error de claridad en los datos.

# Entregables

Se entrega un archivo comprimido vía correo donde se encuentra alojado la aplicación y a su vez la documentación de prueba y carga y los test de métodos del programa, funcionales

# Recursos

## Requerimientos de entornos – Hardware

Router de internet Computador de escritorio 4 gb RAM Unidad de CD/DVD

Servidor local: localhost

## Requerimientos de entornos – Software

Sistema operativo: Windows 10 Codificación: Uso del IDE netbeans 8.0 Xammp

JMeter: entrono de pruebas.

Plataforma de Bootstrap a la cual se realizaron las pruebas

## Herramientas de pruebas requeridas

Las herramientas que se usaron para realizar los test de pruebas y las pruebas de stress y de carga fueron respectivamente:

* JUnit
* JMeter
* Netbeans 8.0
* Navegador Firefox

## Personal

(1) Desarrollador de aplicaciones web con manejo de java

(1) Analista de pruebas

## Entrenamiento

No necesita entrenamiento para el manejo de la aplicación ya que esta es intuitiva y fácil de manejar

# Planificación y organización

## Matriz de responsabilidades

Responsable del proyecto y de las pruebas:

Braham Sebastian Marin Castañeda

## Premisas

Las premisas relacionadas con las tareas de pruebas de software, incluyendo limitaciones de tiempo, disponibilidad de recursos que se asumen, uso de una metodología de pruebas, uso de una herramienta, entre otros.

## Dependencias y Riesgos

* Dependencias con desarrollos.
* Disponibilidad de recursos.
* Premisas que resulten no ser ciertas.

# Glosario

Definiciones de términos usados en la documentación, y general sobre el área de pruebas.

JSTL: La tecnología JavaServer Pages Standard Tag Library es un componente de Java EE. Extiende las ya conocidas Java Server Pages proporcionando cuatro bibliotecas de etiquetas con utilidades ampliamente utilizadas en el desarrollo de páginas web dinámicas.

JDBC template: es la encargada de realizar el trabajo de captura de excepciones, enviar consultas a la base de datos, mapear los resultados (convertir los datos de la tabla en la correspondiente clase), cerrar las conexiones, etc.,

JUnit: conjunto de bibliotecas creadas por Erich Gamma y Kent Beck que son utilizadas en programación para hacer pruebas unitarias de aplicaciones Java

JMeter: JMeter es un proyecto de Apache que puede ser utilizado como una herramienta de prueba de carga para analizar y medir el rendimiento de una variedad de servicios, con énfasis en aplicaciones web.