

UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA
FACULTAD DE INGENIERÍA EN SISTEMAS DE INFORMACIÓN

**PROYECTO FINAL DEL CURSO ASEGURAMIENTO DE CALIDAD DE
SOFTWARE**

**PRUEBAS FUNCIONALES
PRUEBAS NO FUNCIONALES**

Presentado por:

OMAR ALEJANDRO BERMUDEZ CACERES

ALVARO VINICIO ORELLENA MENDEZ

CARLOS EDUARDO CAMEY MILIAN

HERBERT ARAGON MONZON

MAYCOL ARNULFO GUERRA

KEVIN MAURICIO PUMAY GODOY

VILLA NUEVA, OCTUBRE DE 2024

Índice

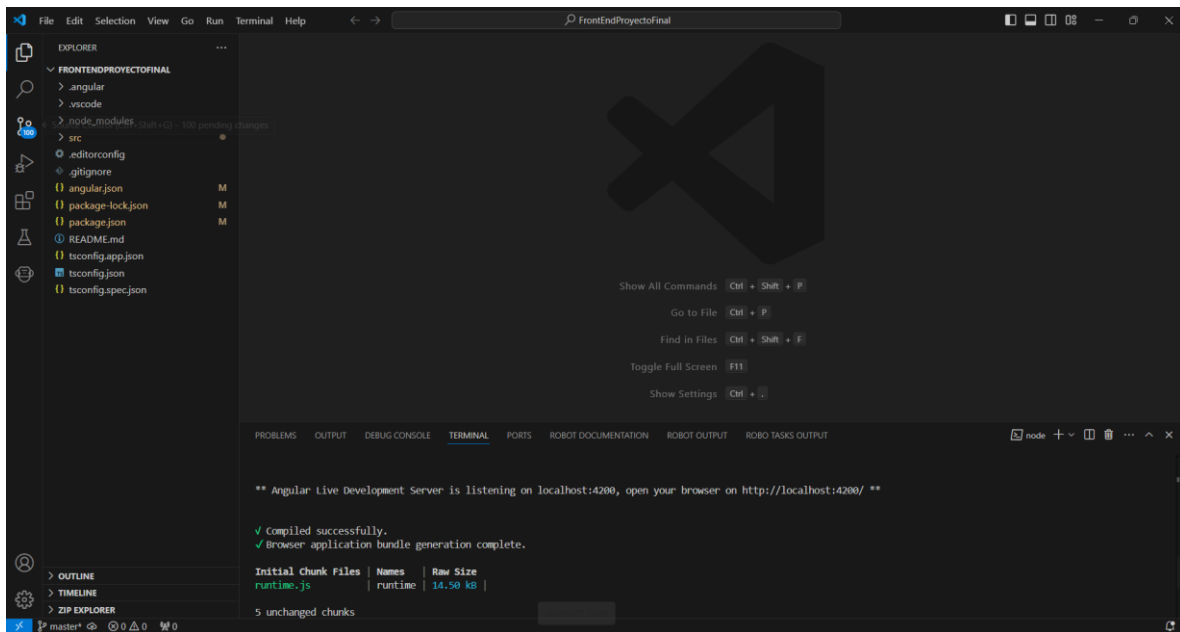
Introducción.....	3
Pruebas Funcionales	4
Pruebas No Funcionales.....	8
Conclusión.....	15

Introducción

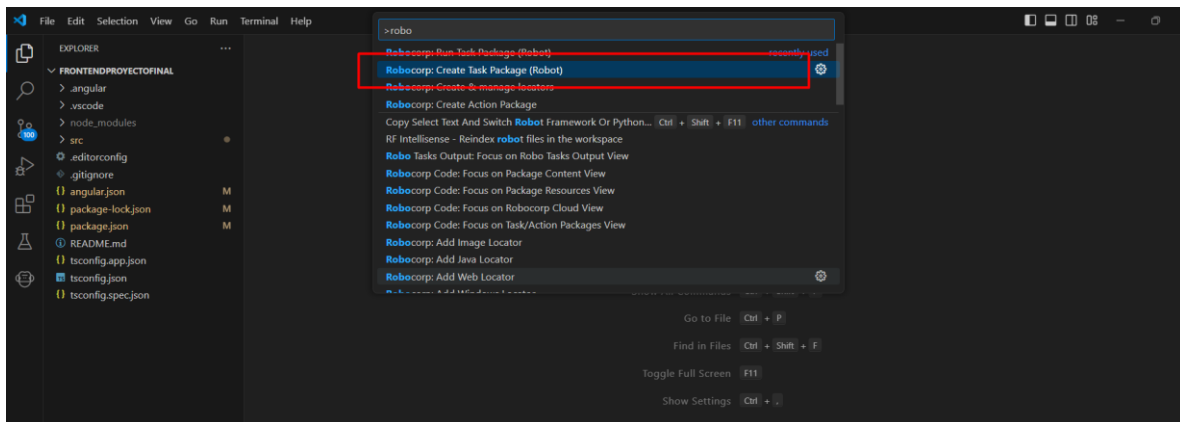
El presente documento tiene como objetivo principal la ejecución de pruebas funcionales y no funcionales en el contexto del aseguramiento de calidad de software. A lo largo de este trabajo, se describen las pruebas realizadas utilizando diversas herramientas, tales como Robot Framework y JMeter, con el fin de evaluar tanto el frontend como el backend de una aplicación web. Las pruebas funcionales se centran en verificar el comportamiento de la aplicación conforme a los requisitos establecidos, mientras que las pruebas no funcionales evalúan aspectos como el rendimiento y la capacidad de respuesta bajo condiciones de estrés. El propósito es asegurar que el sistema cumpla con los estándares de calidad requeridos antes de su implementación definitiva.

Pruebas Funcionales

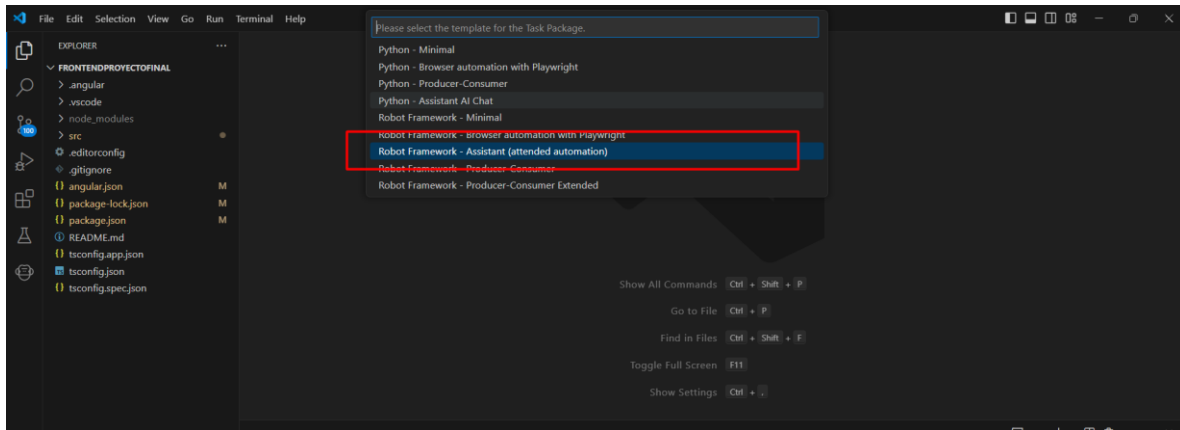
Se hizo las instalaciones de las extensiones de Robot Framework.



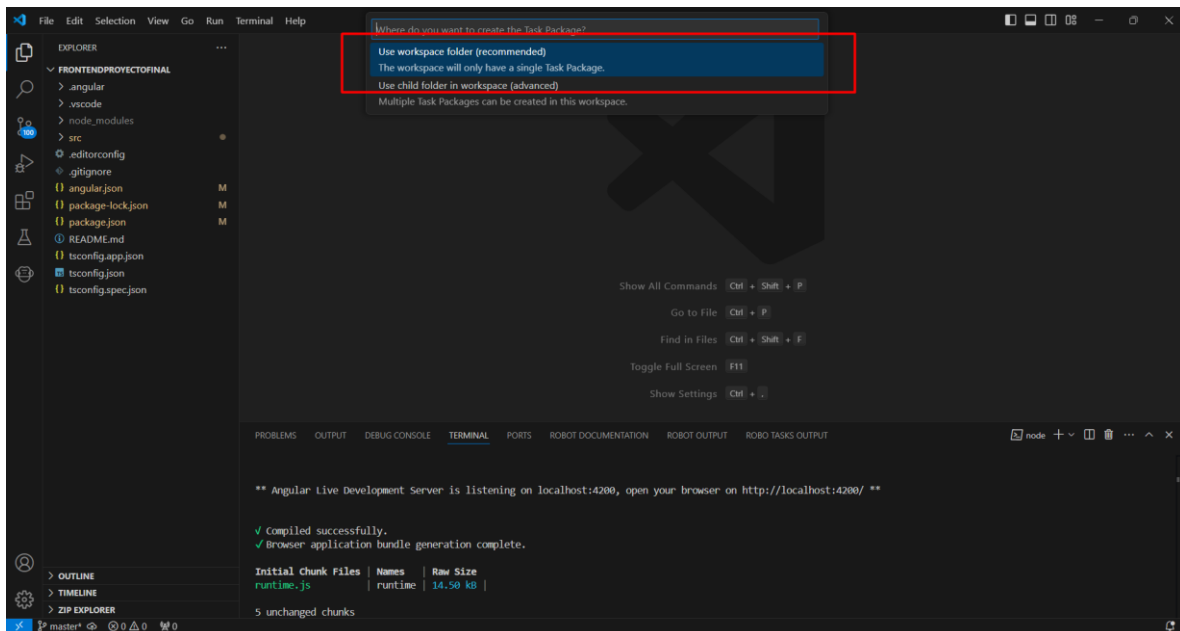
Se hace la creación del Robot Framework para su ejecución dentro de visual studio code (frontend).



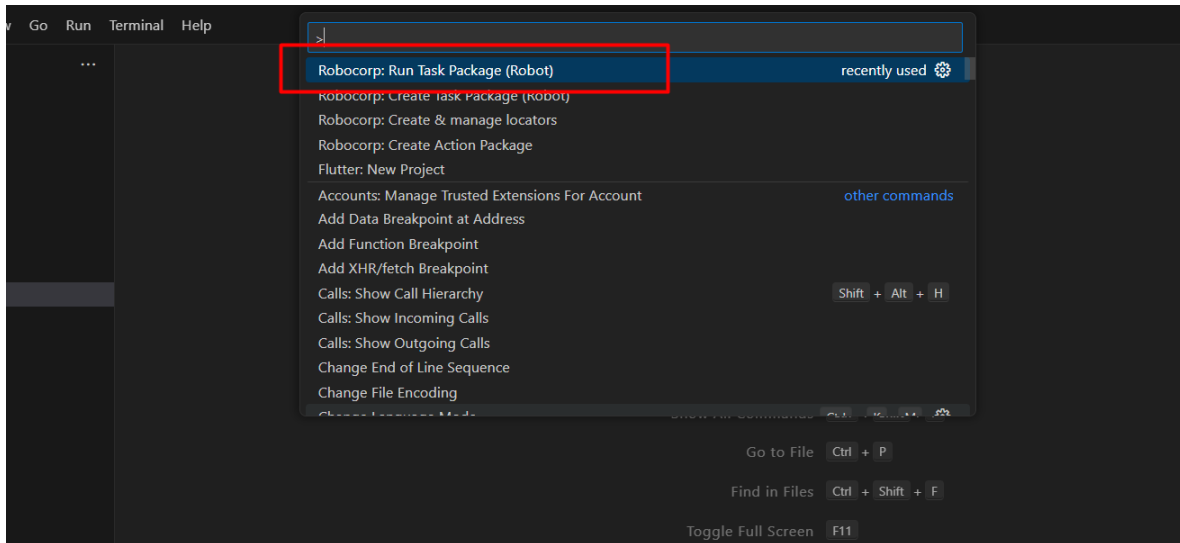
Dentro de las opciones, elegimos la que este más acorde a las necesidades (en este caso asistente).



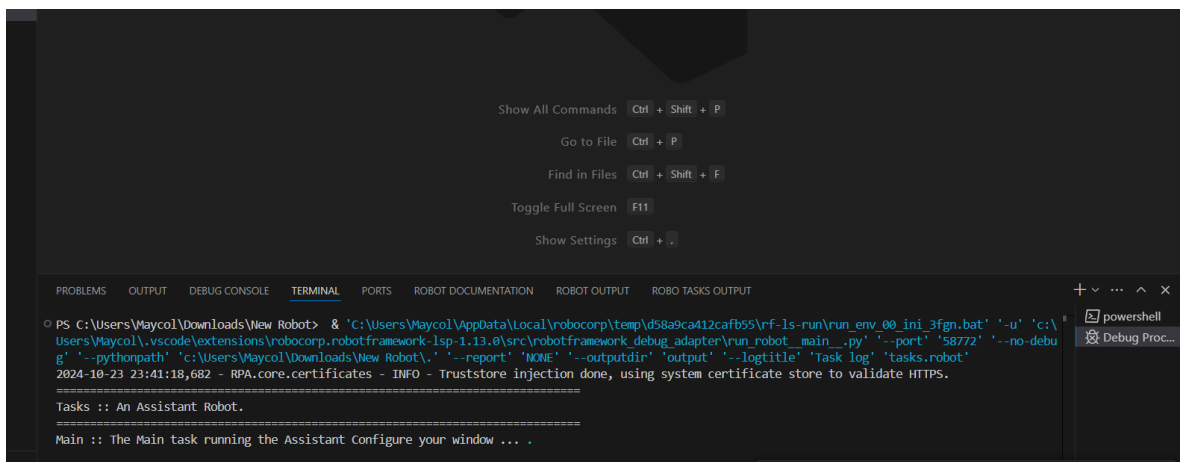
Elegimos la opción de “Use Workspace folder” para que se creen las instancias en nuestro código para evaluar sus funcionalidades.



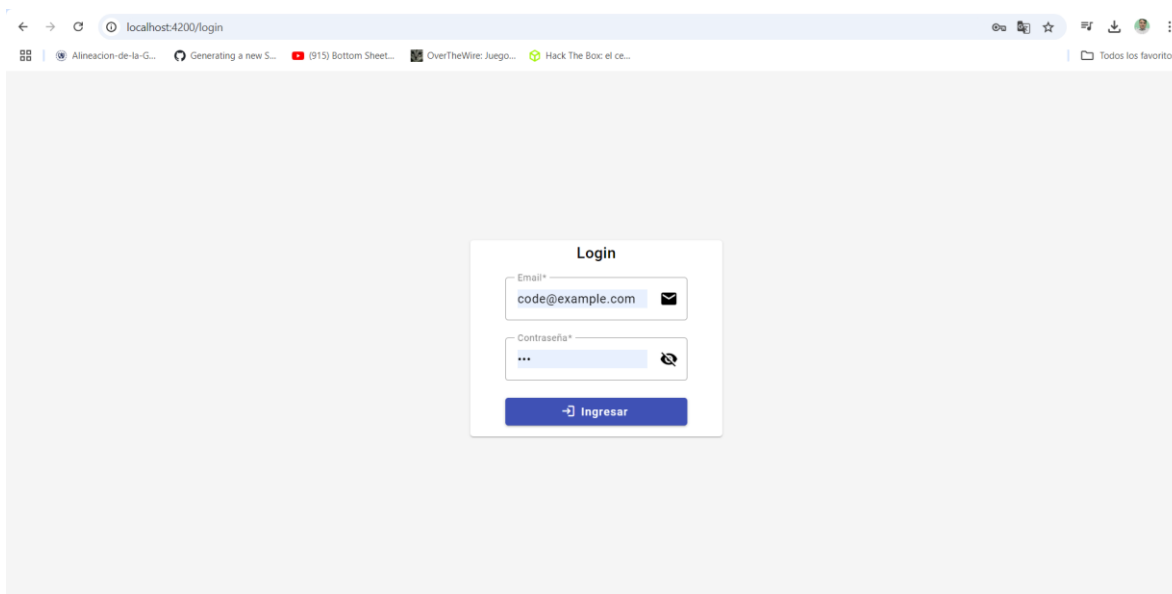
Iniciamos el proceso con “Robor: Run”.



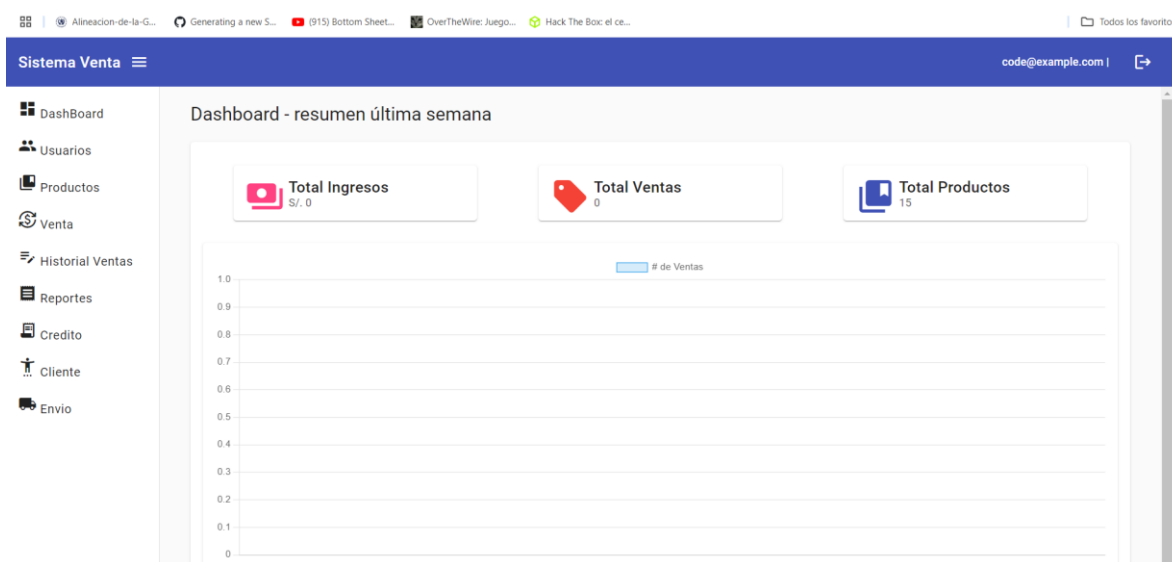
Se corre el proceso, esperamos que se cargue (cuando es primera vez se descargan archivos complementarios).



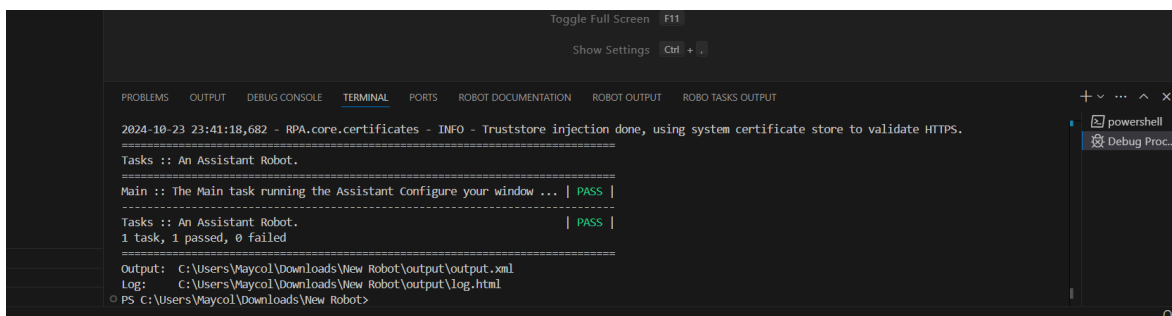
Se cargará y abrirá la ejecución del código.



Procedemos a ejecutar la página mientras el analiza las situaciones.

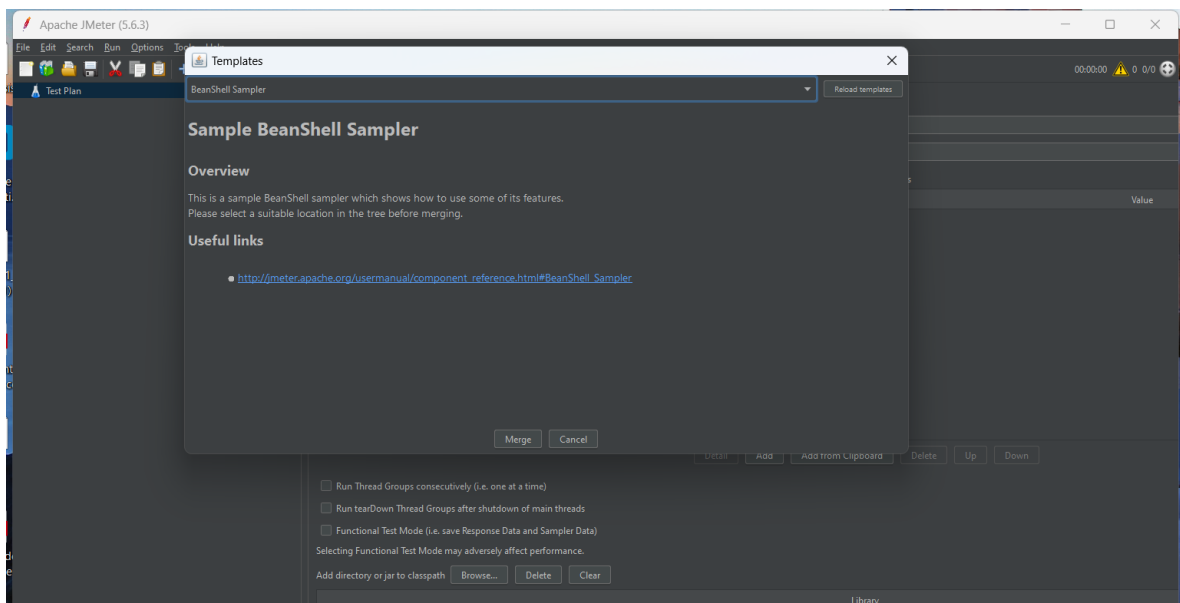
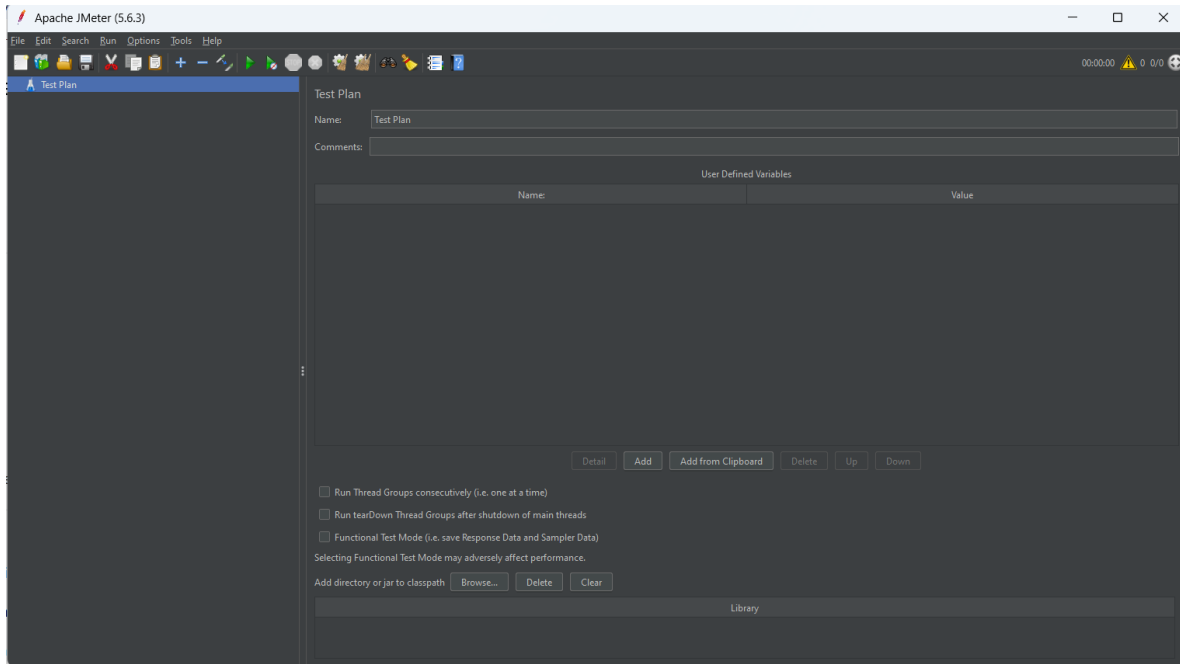


Cuando finaliza el proceso, se cierra y brinda si ha sido exitoso el análisis.

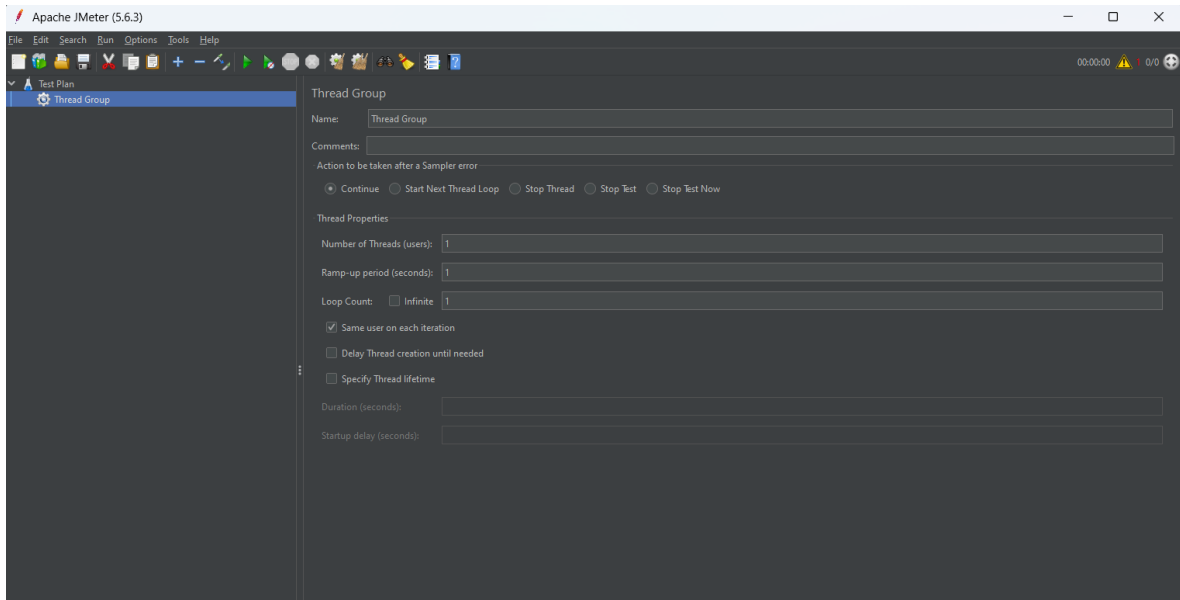


Pruebas No Funcionales

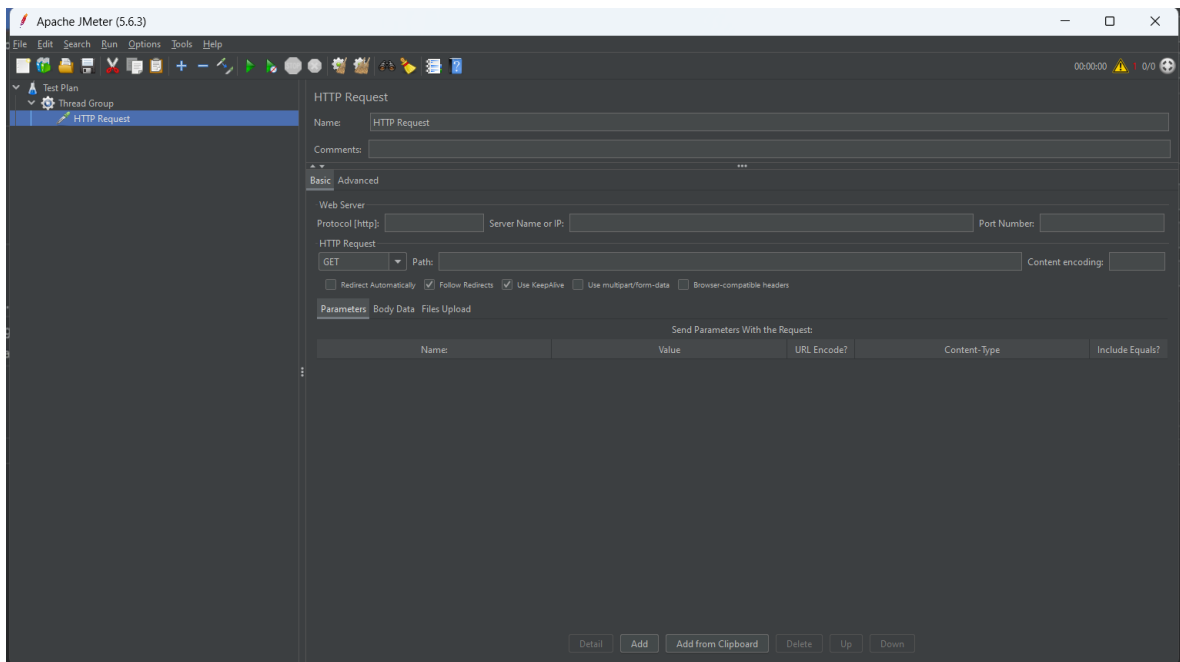
Para realizar estas pruebas se necesitará la herramienta “Jmeter” la cual pondrá a prueba nuestro código de forma estresándolo (también depende del equipo con que se corra).



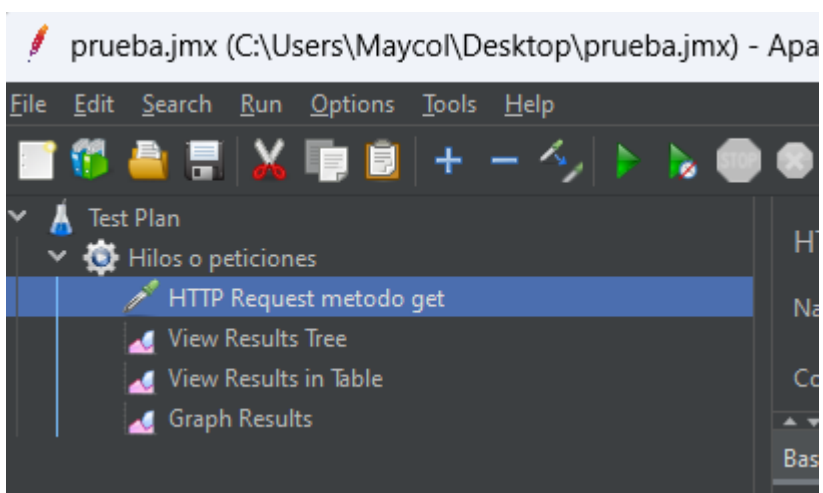
Empezamos agregando Hilos peticiones (Thread Group).



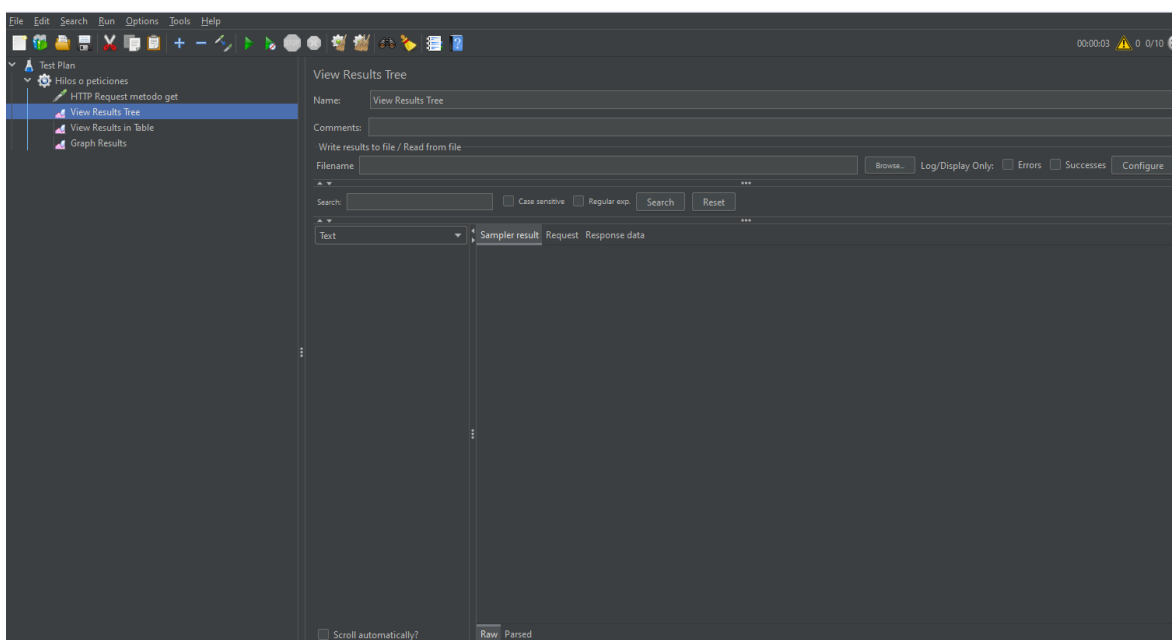
Agregaremos “HTTP Request” para lograr hacer conexión con nuestra página y poner a prueba.



También agregaremos las opciones de: View Results Tree, View Results in Table, Graph Results.



Estas opciones servirán para observar un gráfico, verificar si ha realizado conexión



Verificamos nuestra página y verificamos lo siguiente:

<http://localhost:4200/pages/dashboard>

Protocolo

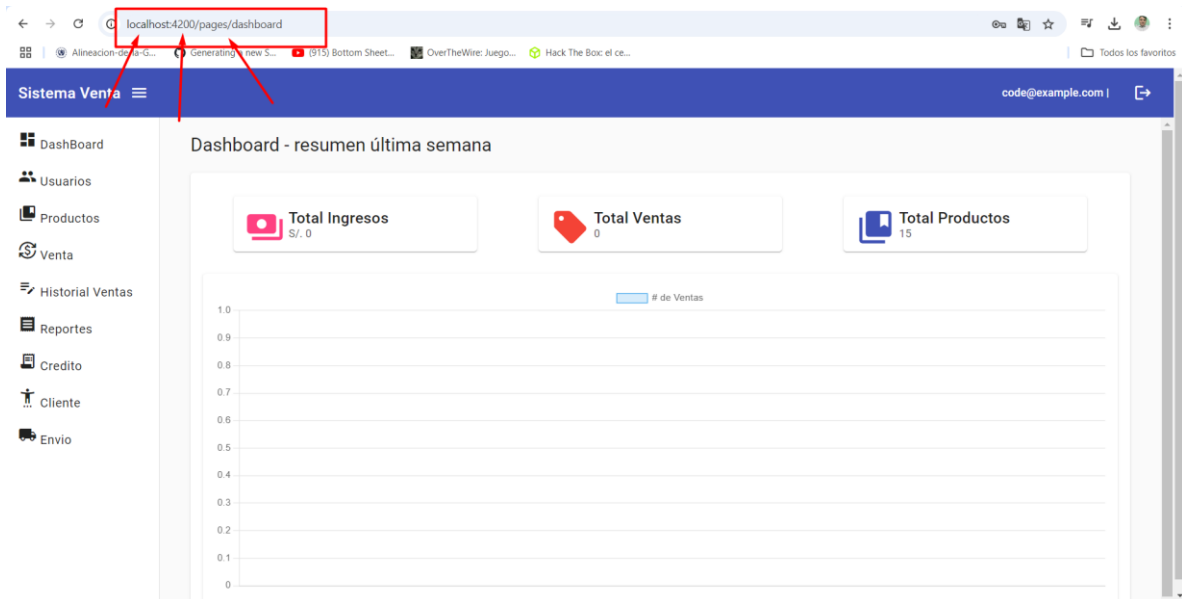
Servidor

Puerto

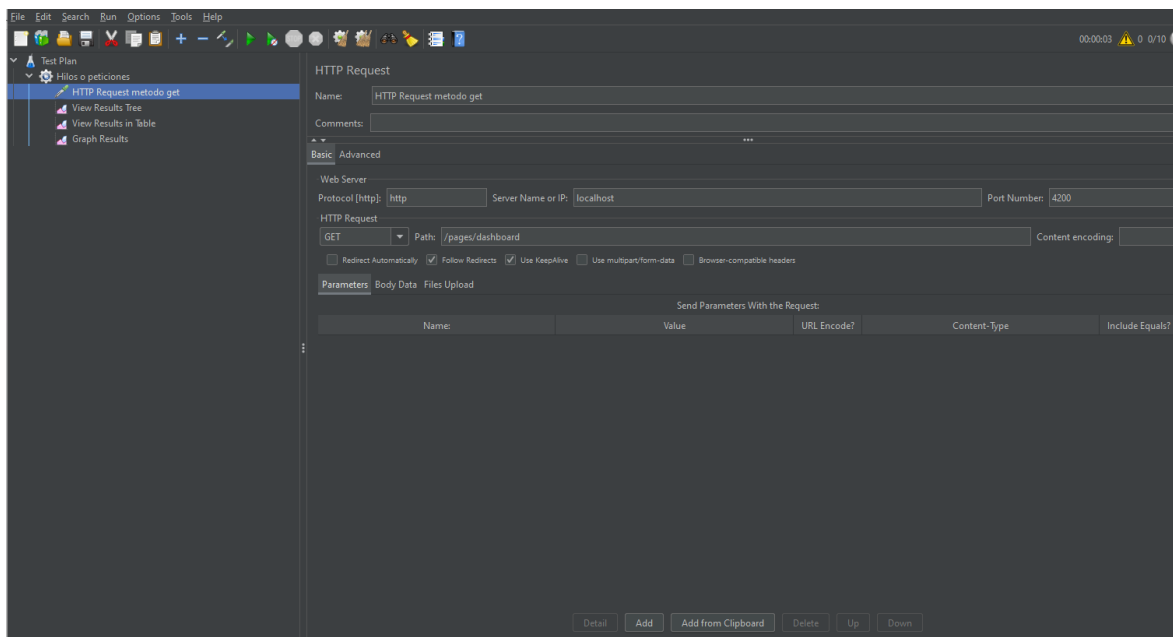
HTTP Request

Paht

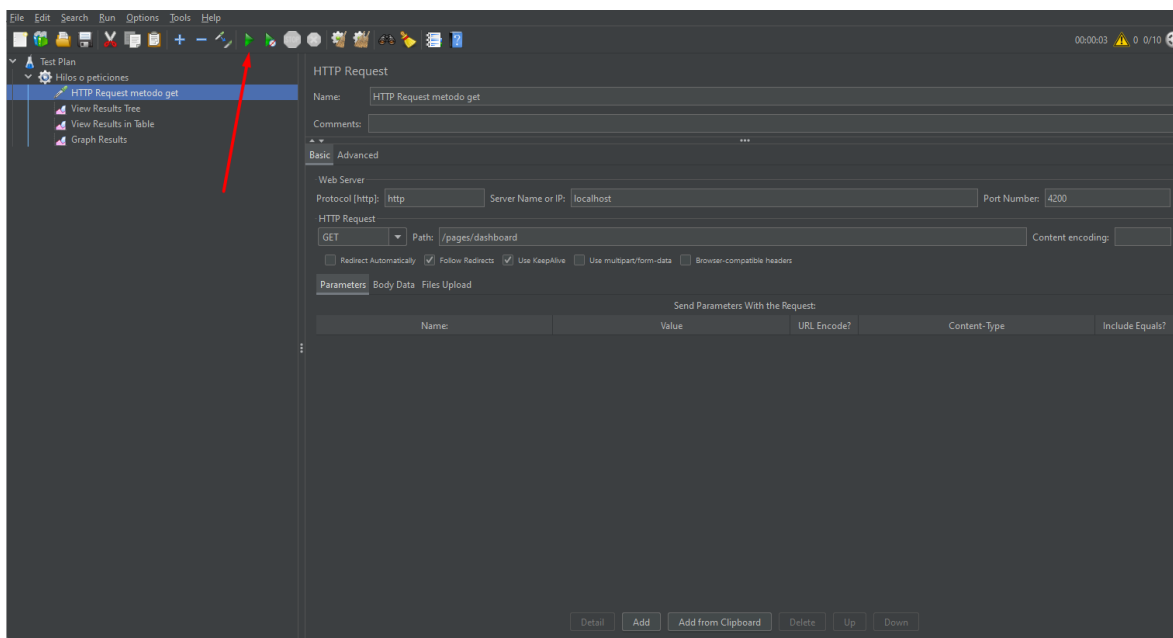
Esta información es importante ya que aquí es donde dirigiremos el Jmeter hacia la página que pondremos a prueba.



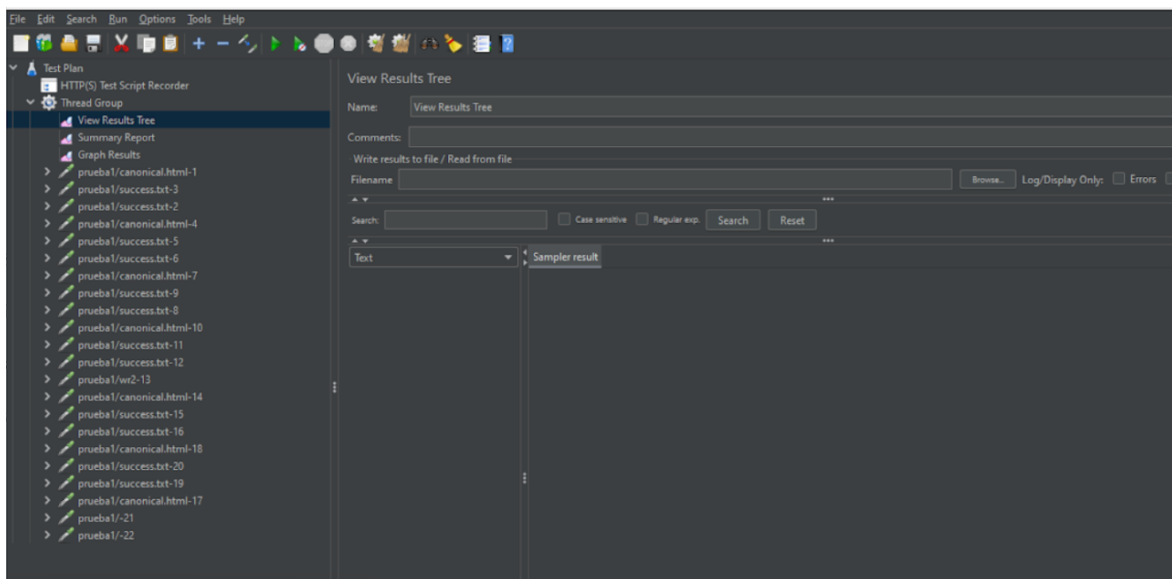
Se procede a llenar los campos mencionados al Jmeter.



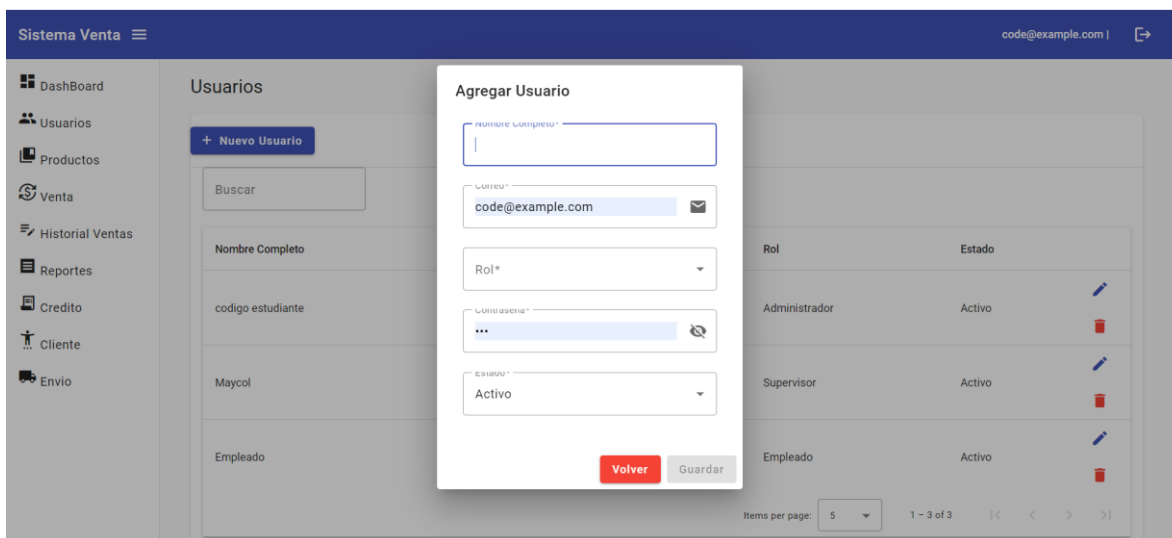
Se presiona el botón verde para iniciar la prueba.



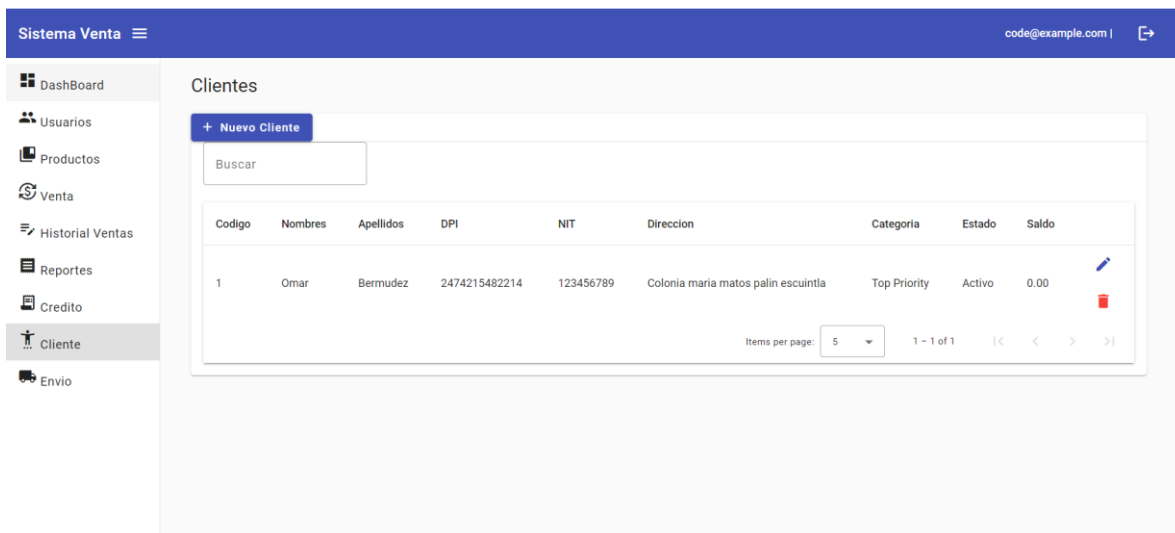
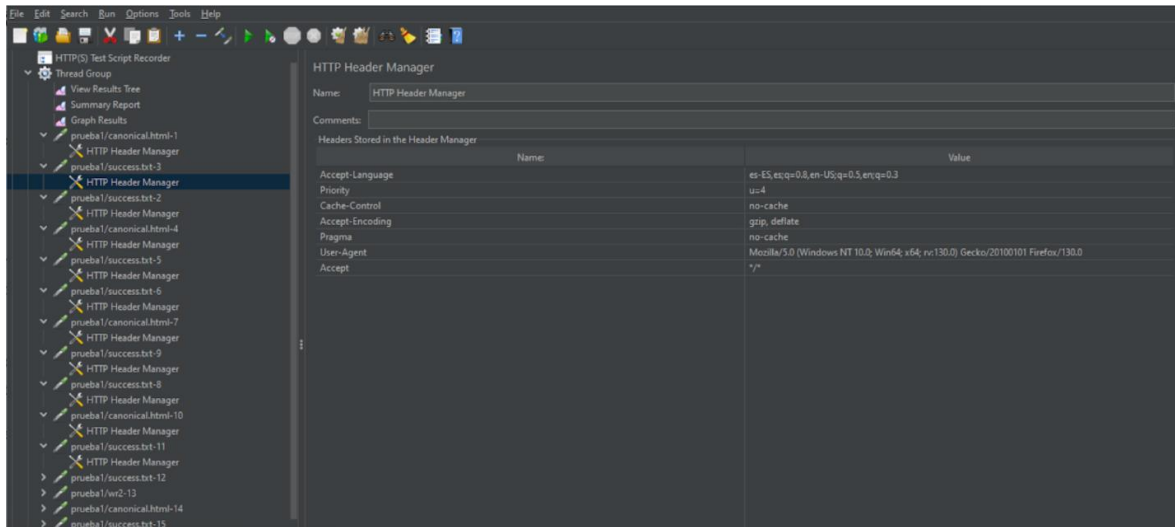
Se va observando cómo se pone a prueba la página mientras es utilizada.



Usamos la página web para que Jmeter detecte zonas para poner a prueba la pagina web.



Mientras se usa la página web se valida como ha funcionado el Jmeter.



Conclusión

En conclusión, las pruebas funcionales y no funcionales realizadas demostraron la eficacia de los métodos y herramientas implementadas para garantizar la calidad del software. Las pruebas funcionales, ejecutadas mediante Robot Framework, confirmaron que la aplicación cumple con las funcionalidades requeridas, mientras que las pruebas no funcionales, realizadas con JMeter, evidenciaron la capacidad del sistema para soportar distintas cargas de trabajo. Estos resultados permiten asegurar que la aplicación está en condiciones óptimas para ser utilizada, garantizando un rendimiento adecuado y una experiencia de usuario satisfactoria.