**API GA7-220501096-AA5-EV02.**

Fase: Ejecución

Actividad No.5

Evidencia No.2

Por:

Alexander Junco Herrera

Servicio Nacional de Aprendizaje SENA

Tecnólogo en Análisis y Desarrollo de Software

Ficha:2853234

Instructor: Milton Ivan Barbosa Gaona

9 mayo de 2025

Tabla de Contenido

[Justificación 3](#_Toc197012700)

[Introducción 4](#_Toc197012701)

[Objetivos 5](#_Toc197012702)

[Conclusión 14](#_Toc197012703)

[Bibliografía 15](#_Toc197012704)

# Justificación

La correcta validación de una API garantiza que los servicios ofrecidos por una aplicación funcionen de forma adecuada, segura y eficiente. Postman es una herramienta que permite realizar pruebas de endpoints HTTP, proporcionando retroalimentación inmediata del comportamiento del servidor frente a diferentes solicitudes. En el marco del proyecto **JF Zambrano Call Center**, esta evidencia permite demostrar el correcto funcionamiento de los endpoints implementados mediante el uso de esta herramienta. Además, se asegura que los flujos de autenticación y servicios cumplan con los requerimientos del sistema propuesto.

# Introducción

En esta evidencia se documenta el proceso de testing realizado sobre la API del proyecto **JF Zambrano Call Center**, utilizando la herramienta Postman. Se detallan los pasos seguidos para instalar Postman, configurar y enviar solicitudes HTTP a cada uno de los endpoints definidos en el sistema, y se incluyen capturas de pantalla que muestran los resultados obtenidos. Esta actividad permite validar el correcto funcionamiento de los servicios como inicio de sesión, validación de credenciales y visualización de servicios ofrecidos por la empresa. Además, se elabora un video demostrativo y se entrega toda la estructura del proyecto como parte del producto final.

# Objetivos

**Objetivo general:**

Realizar pruebas funcionales a los endpoints del proyecto JF Zambrano Call Center utilizando la herramienta Postman, con el fin de validar el correcto comportamiento de las solicitudes y respuestas HTTP en la API desarrollada.

**Objetivos específicos:**

1. Instalar y configurar correctamente la herramienta Postman para realizar pruebas sobre los endpoints del proyecto.
2. Ejecutar pruebas de tipo GET y POST sobre los endpoints definidos para validar el flujo de autenticación y acceso a recursos.
3. Documentar mediante capturas de pantalla y video los resultados obtenidos en las pruebas, explicando los códigos de estado HTTP y su significado.

**Evidencia de producto: GA7-220501096-AA5-EV02 API**

Teniendo en cuenta lo realizado en la AA5-EV01 (evidencia anterior) realice el testing de la API construida usando la herramienta postman.

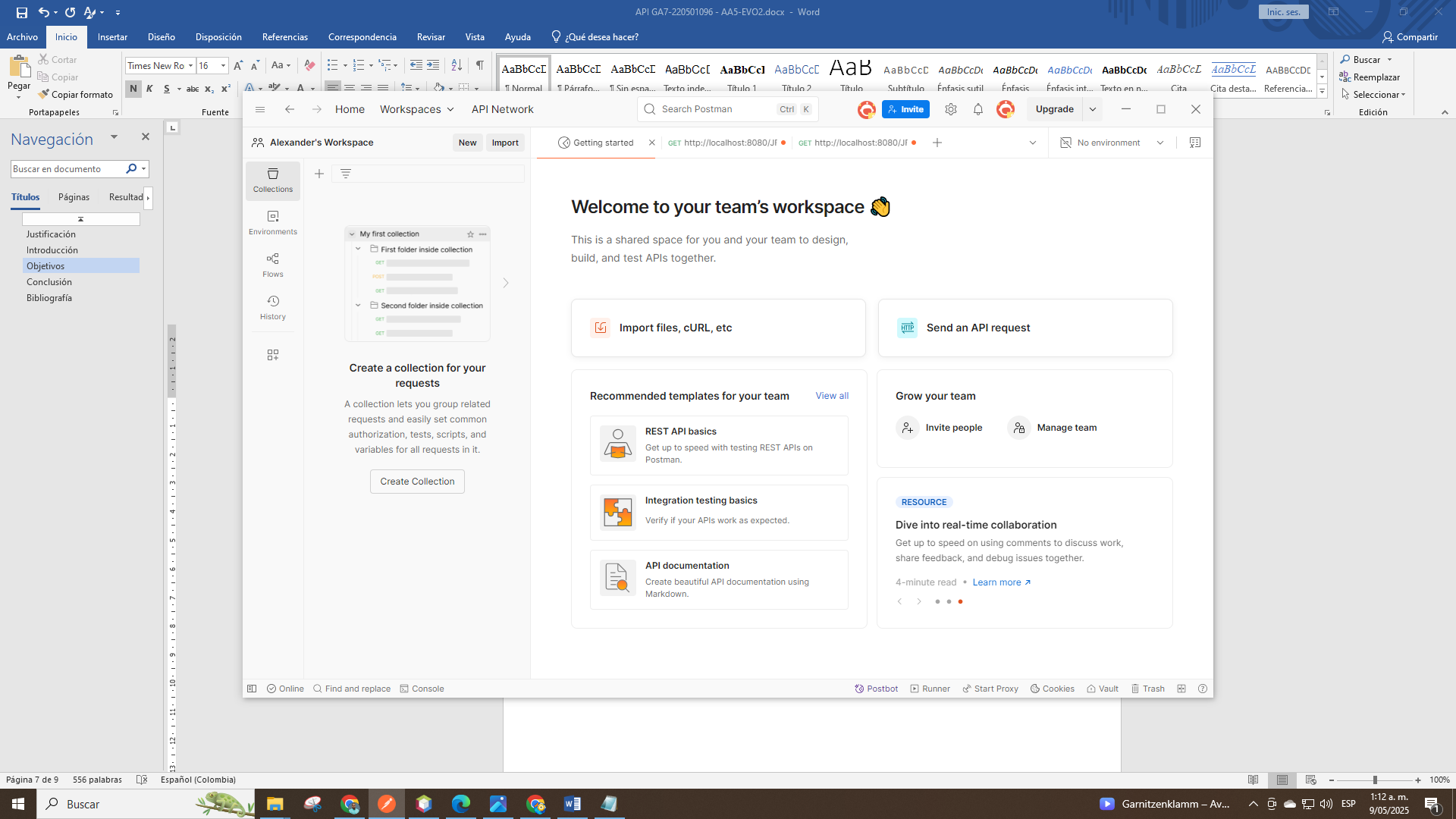
**Elementos para tener en cuenta:**

* Debe instalar la herramienta postman.
* Debe realizar un video mostrando el testing de la API con la herramienta Postman.
* Documentar con pantallazos el testing realizado.
* Entregar los ENDPOINT de las API´s

**Lineamientos generales para la entrega de la evidencia:**

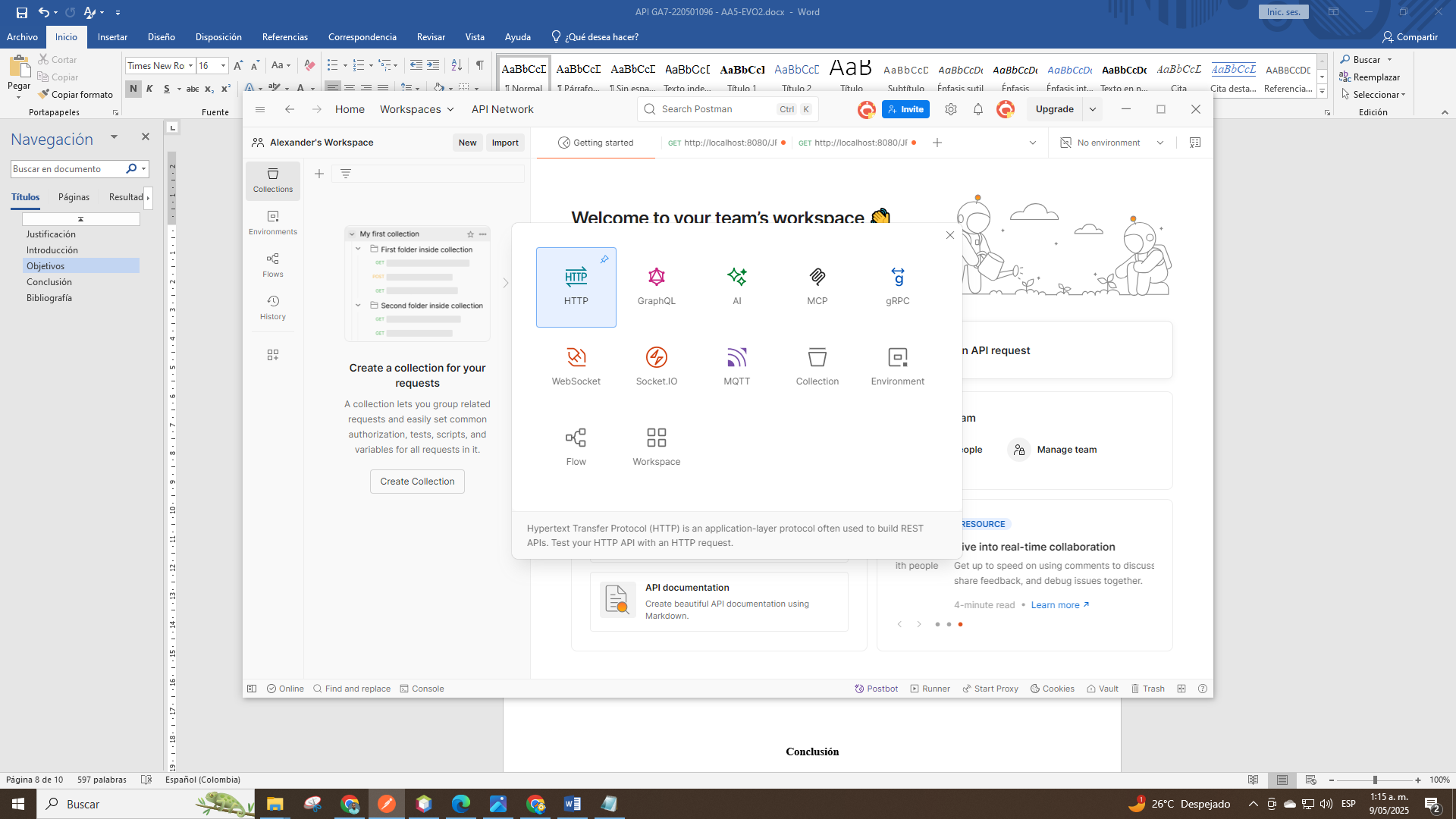
* **Productos para entregar:** carpeta comprimida que debe tener los siguientes archivos:
* Archivos del proyecto,
* Video
* Documento con pruebas.
* Archivo END POINTS.
* Archivo con enlace del repositorio, la carpeta comprimida debe tener el nombre del aprendiz y número de la evidencia así: NOMBRE\_APELLIDO\_AA5\_EV02
* **Extensión:** ZIP, RAR.
* Para hacer el envío de la evidencia remítase al área de la actividad correspondiente y acceda al espacio: **API GA7-220501096-AA5-EV02.**

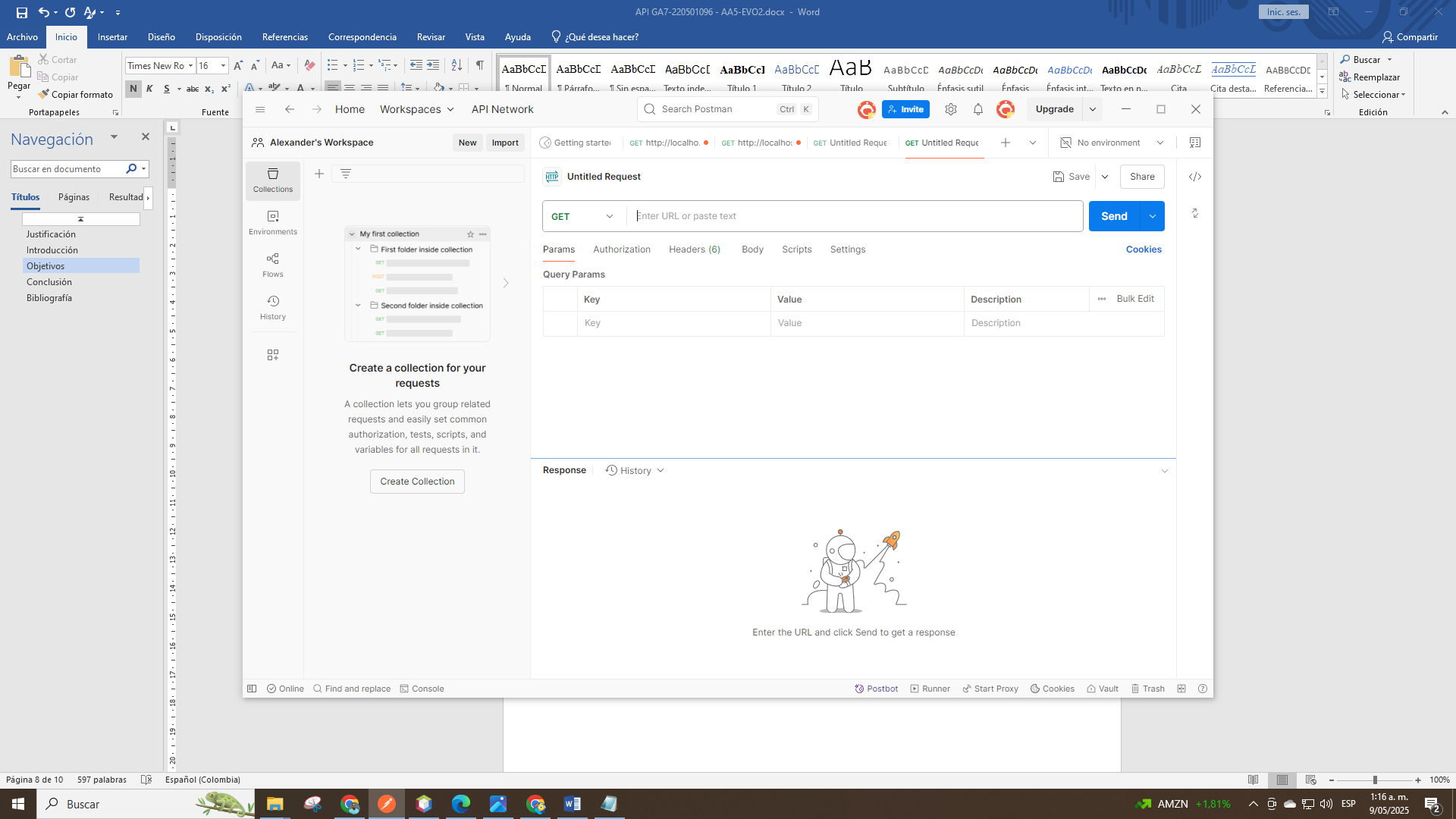
**📥 1. Descargar e instalar Postman**

1. Ve al sitio web oficial: https://www.postman.com/downloads/
2. Descarga la versión según tu sistema operativo (Windows, macOS o Linux).
3. Instala y abre el programa.
4. (CAPTURA #1) **Pantalla de inicio de Postman (Dashboard)**.

**🧪 2. Crear una nueva solicitud HTTP**

1. Haz clic en el botón **"New"** (parte superior izquierda).
2. Selecciona **"HTTP Request"**.
3. Se abrirá una nueva pestaña donde podrás configurar tu prueba.
4. (CAPTURA #2) **Pantalla de creación de nueva solicitud**.





**🔗 3. Probar el endpoint inicio\_sesion.jsp**

* **Método:** GET
* **URL:**

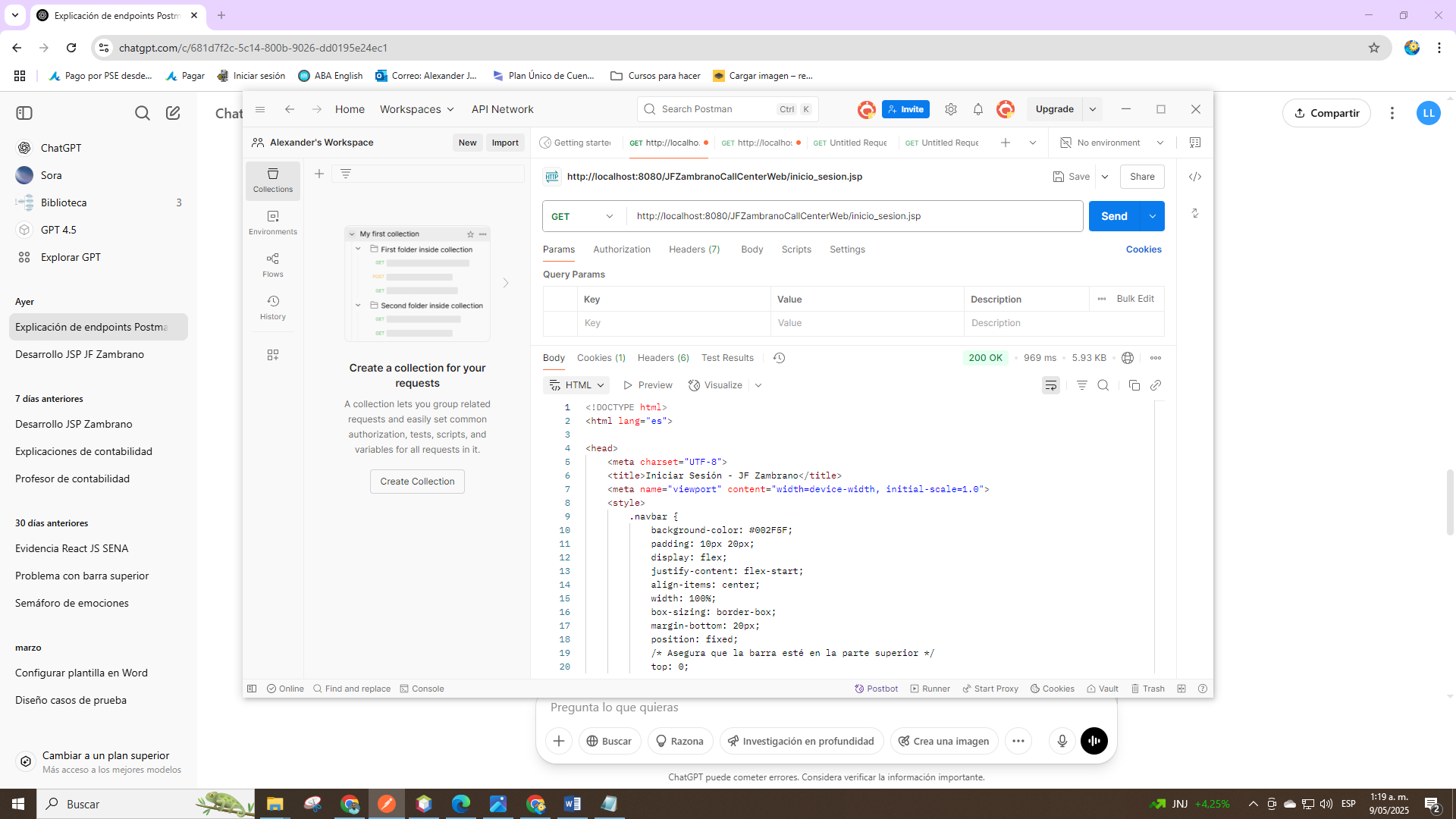
bash

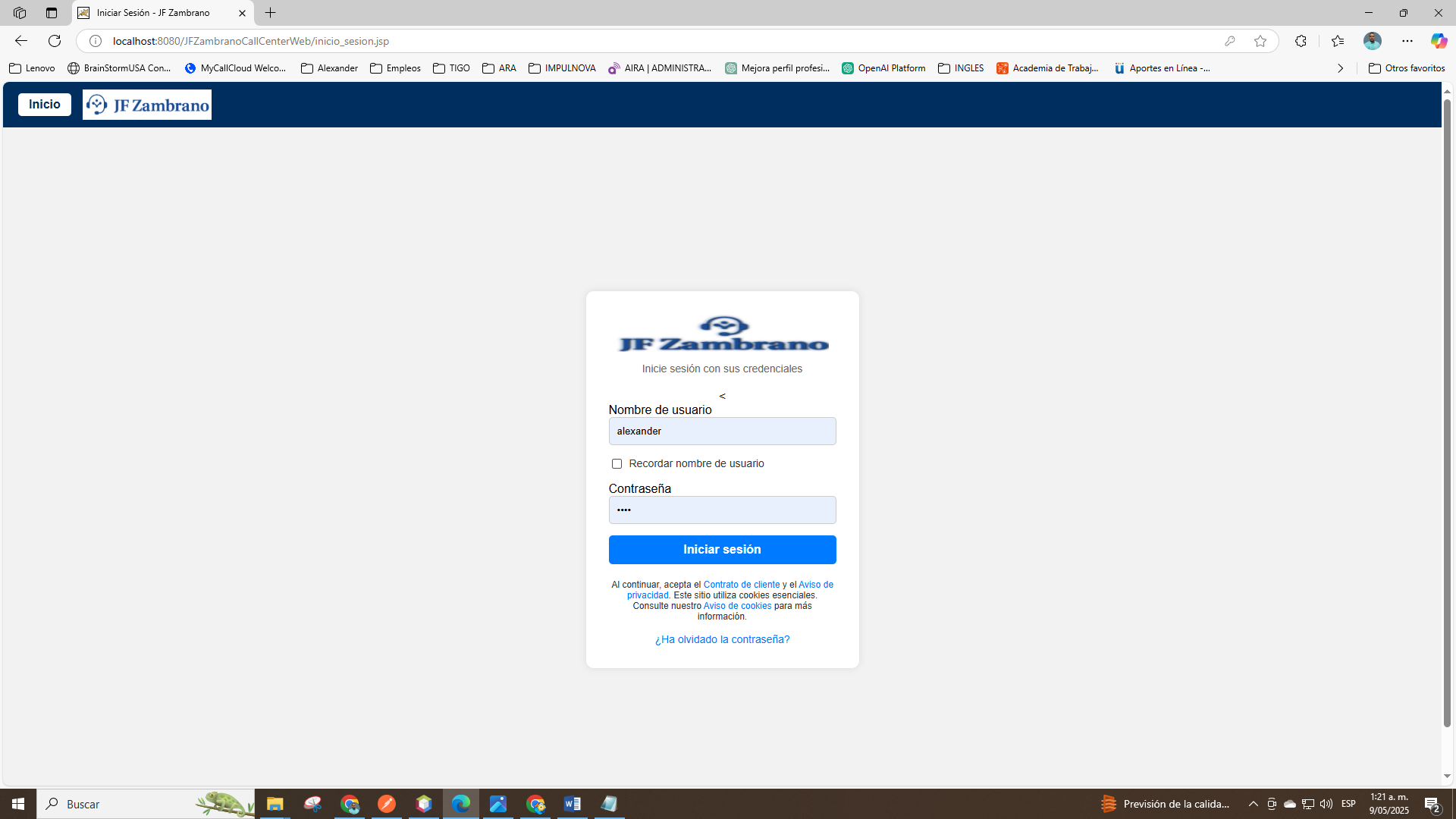
CopiarEditar

http://localhost:8080/JFZambranoCallCenterWeb/inicio\_sesion.jsp

* Haz clic en **"Send"**
* Deberías ver el código **200 OK** si todo está bien.

📸 (CAPTURA #3)

* Método GET, URL colocada, botón Send presionado.
* Resultado en la parte inferior mostrando **Status: 200 OK**.



**❌ 4. Probar login con contraseña incorrecta**

* **Método:** POST
* **URL:**

bash

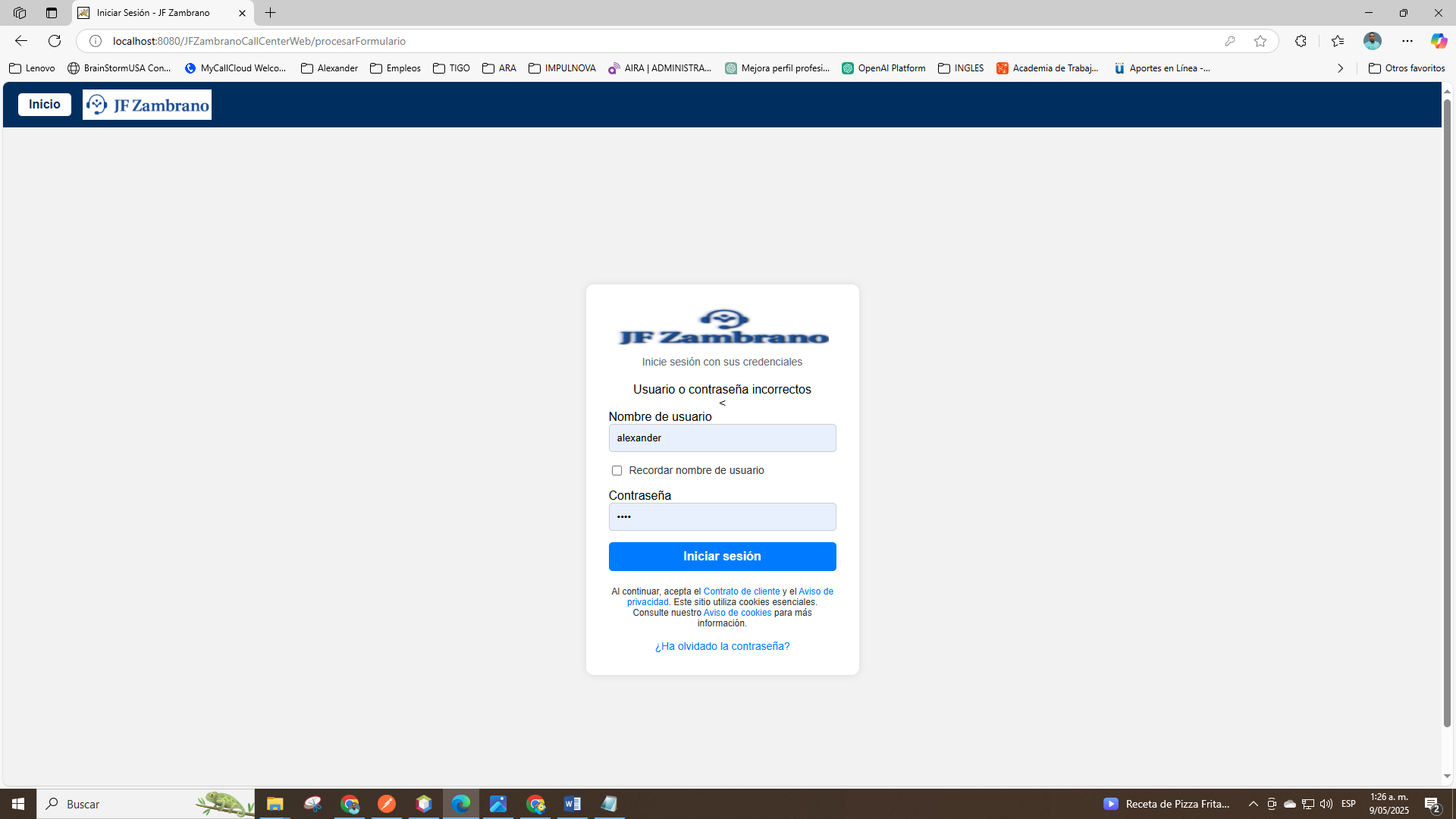
CopiarEditar

http://localhost:8080/JFZambranoCallCenterWeb/procesarFormulario

* Ve a la pestaña **Body → x-www-form-urlencoded**
* Llena los campos:
  + usuario: admin
  + password: errada
* Haz clic en **"Send"**

📸 (CAPTURA #4)

* Cuerpo del formulario con datos incorrectos, Status diferente (ej. 401 o mensaje de error personalizado).



**✅ 5. Probar login con contraseña correcta**

* Mismo método: POST
* Misma URL:

bash

CopiarEditar

http://localhost:8080/JFZambranoCallCenterWeb/procesarFormulario

* En **Body → x-www-form-urlencoded**, escribe:
  + usuario: admin
  + password: 1234
* Haz clic en **"Send"**

📸 (CAPTURA #5)

* Resultado exitoso mostrando redirección o HTML de bienvenida.
* Status: **200 OK**.

**📋 6. Probar la página de servicios**

* **Método:** GET
* **URL:**

bash

CopiarEditar

http://localhost:8080/JFZambranoCallCenterWeb/servicios.jsp

* Haz clic en **"Send"**

📸 (CAPTURA #6)

* Resultado muestra HTML de la página de servicios
* Status: **200 OK**

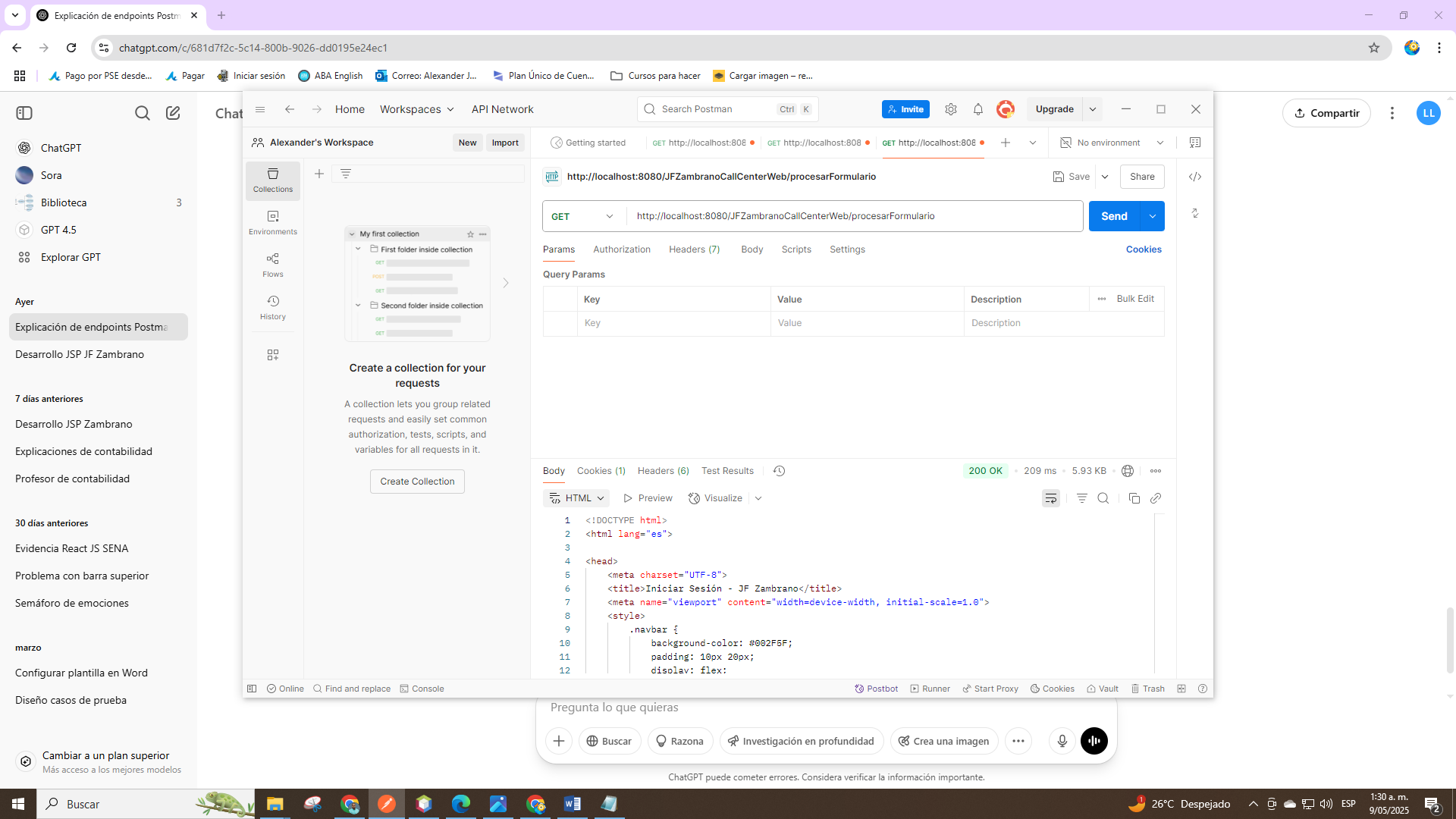
**📌 7. Explicación del código 200 OK**

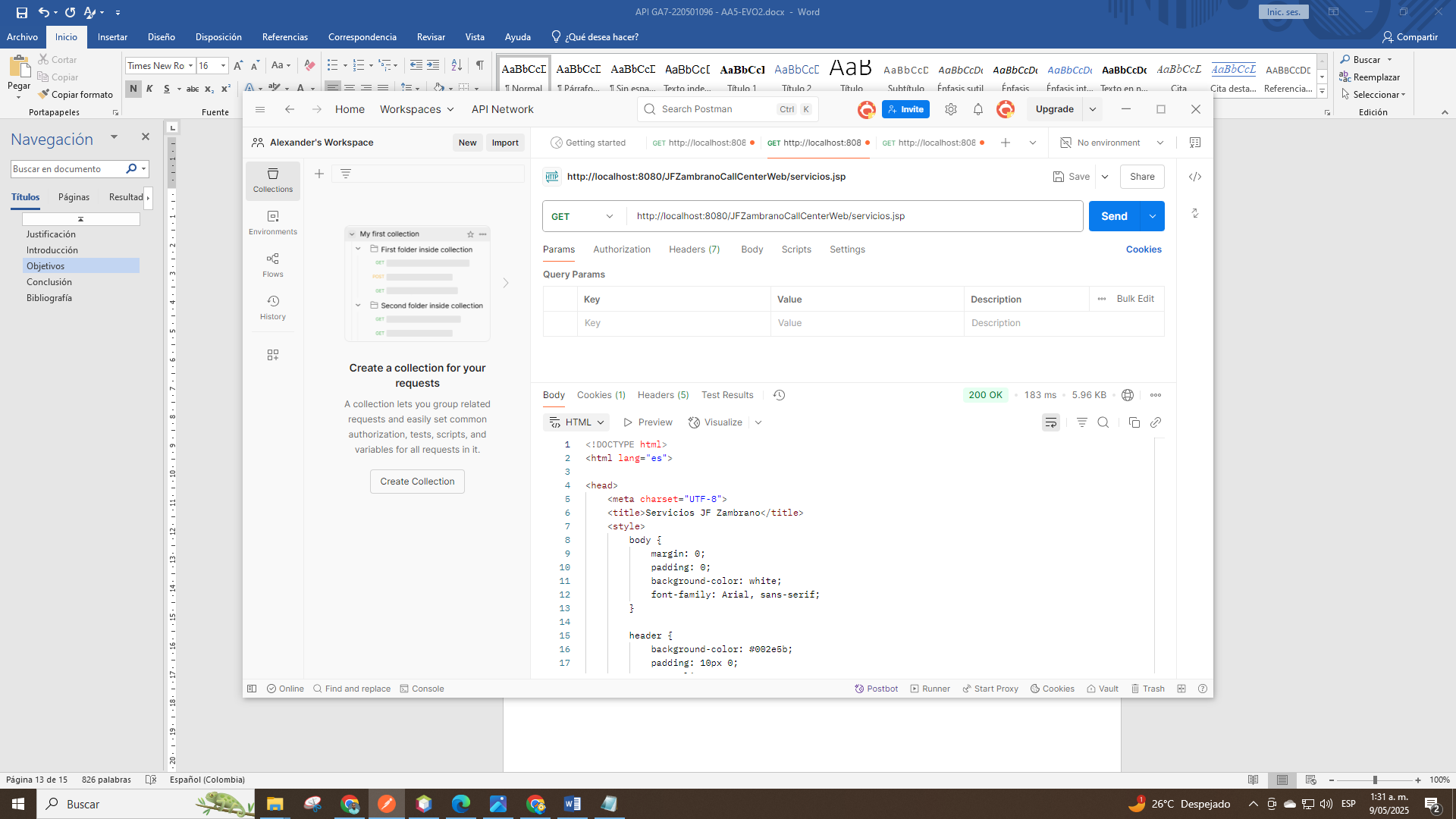
Cada vez que Postman responde con **200 OK**, significa que la solicitud fue recibida, comprendida y procesada correctamente por el servidor.

Puedes decir esto en tu trabajo:

"El código 200 OK indica que el servidor ejecutó con éxito la solicitud enviada desde Postman. Es un indicador de funcionamiento correcto del endpoint."

**Inicio**



**Servicios**

# Conclusión

1. Se logró instalar y utilizar adecuadamente la herramienta Postman, lo cual permitió generar solicitudes HTTP de forma controlada y segura para el testing de la API.
2. Los endpoints del sistema respondieron correctamente a las solicitudes GET y POST, evidenciando un funcionamiento estable en los procesos de inicio de sesión y acceso a información.
3. La documentación en video y en imágenes permitió reflejar de forma clara la utilidad de los códigos de estado como el **200 OK**, y dejó constancia de que la API responde correctamente a escenarios positivos y negativos.

# Bibliografía

* *LMS Zajuna - "* *Desarrollo de Front-end con REACT JS”* <https://zajuna.sena.edu.co/Repositorio/Titulada/institution/SENA/Tecnologia/228118/Contenido/OVA/CF31/index.html#/curso/tema1>
* Facebook. (2023). React – A JavaScript library for building user interfaces. <https://reactjs.org/>
* W3Schools. (2023). React JS Tutorial. <https://www.w3schools.com/react/>