**UNVERSIDAD NACIONAL DEL ALTIPLANO**

**UNIDAD DE POSGRADO FACULTAD DE MECANICA ELÉCTRICA, ELECTRÓNICA Y SISTEMAS**

**MAESTRIA EN INGENIERIA DE SISTEMAS**

****

LABORATORIOS

**PRESENTADO POR**

ING. GOMEZ ALCOS ELMER VALERIO

**CURSO**

SISTEMAS DISTRIBUIDOS Y PARALELOS

**DOCENTE**

DR. LENIN HUAYTA FLORES

PUNO – 2025

**INDICE**

[INFORME DE LABORATORIOS 3](#_Toc199689461)

[INTRODUCCIÓN 3](#_Toc199689462)

[Laboratorio 01: Rendimiento (Comparación) con Joblib 3](#_Toc199689463)

[Laboratorio 02: Cliente-Servidor con Sockets 3](#_Toc199689464)

[Laboratorio 03: RMI en Java 4](#_Toc199689465)

[Laboratorio 04: Ejemplo con Hilos y Canvas en Java 4](#_Toc199689466)

[Laboratorio 05: JSP con Apache Tomcat 4](#_Toc199689467)

[Laboratorio 06: JavaBean en JSP 4](#_Toc199689468)

[Laboratorio 07: Reserva en JSP 5](#_Toc199689469)

[Laboratorio 08: Gestión Académica en Django 5](#_Toc199689470)

[CONSLUSIONES 6](#_Toc199689471)

[CAPTURAS 7](#_Toc199689472)

[LABORATORIO 01 7](#_Toc199689473)

[LABORATORIO 02 8](#_Toc199689474)

[LABORATORIO 03 9](#_Toc199689475)

[LABORATORIO 04 11](#_Toc199689476)

[LABORATORIO 05 12](#_Toc199689477)

[LABORATORIO 06 13](#_Toc199689478)

[LABORATORIO 07 14](#_Toc199689479)

[LABORATORIO 08 16](#_Toc199689480)

# INFORME DE LABORATORIOS

**Curso : Sistemas Distribuidos**

**Alumno: Elmer Valerio Gómez Alcos  
Repositorio de los códigos:**

[**https://github.com/ValerioGomez/lab-sistemas-distribuuidos**](https://github.com/ValerioGomez/lab-sistemas-distribuuidos)

# INTRODUCCIÓN

Este informe presenta un resumen detallado de los laboratorios realizados durante el curso de Sistemas Distribuidos. Cada laboratorio abordó aspectos fundamentales de la programación distribuida y concurrente, aplicando tecnologías y herramientas actuales como Python, Java, JSP, Apache Tomcat, Django y paradigmas de concurrencia, con el objetivo de fortalecer las competencias en diseño e implementación de sistemas distribuidos.

## Laboratorio 01: Rendimiento (Comparación) con Joblib

**Proyecto alojado en:** [Google Colab](https://colab.research.google.com/drive/1GIXZIhORfLkdlICe61dPXtxQcdCwkEjg?usp=sharing)

**Descripción:**  
Se realizó una comparación del rendimiento entre estructuras de datos nativas de Python (listas) y arreglos de NumPy, además de experimentar con paralelización utilizando la librería Joblib.

**Logros:**

* Se utilizó la función time.time() para medir los tiempos de ejecución con precisión.
* Se compararon tiempos de procesamiento entre listas nativas y arreglos de NumPy, demostrando que NumPy ofrece un rendimiento superior.
* Se implementó paralelización con Joblib, variando el número de núcleos y el tamaño del lote de tareas.
* Se concluyó que la paralelización no siempre mejora el rendimiento, especialmente en operaciones simples o altamente optimizadas como las que realiza NumPy.

## Laboratorio 02: Cliente-Servidor con Sockets

**Herramienta:** NetBeans

**Descripción:**  
Se implementó una aplicación básica de comunicación cliente-servidor utilizando sockets en Java, estableciendo conexión, envío y recepción de mensajes entre ambos extremos.

**Logros:**

* Desarrollo e implementación exitosa de un modelo cliente-servidor funcional.
* Comprensión práctica del manejo de conexiones TCP/IP con sockets en Java.

## Laboratorio 03: RMI en Java

**Herramienta:** NetBeans

**Descripción:**  
Se implementó un sistema distribuido utilizando Remote Method Invocation (RMI) en Java, para invocar métodos remotos y compartir funcionalidad entre aplicaciones cliente y servidor.

**Logros:**

* Configuración y desarrollo de una aplicación RMI funcional.
* Comprensión del uso de interfaces remotas y manejo de objetos distribuidos en Java.

## Laboratorio 04: Ejemplo con Hilos y Canvas en Java

**Herramienta:** NetBeans

**Descripción:**  
Se desarrolló una aplicación gráfica en Java que utilizó hilos (threads) para animar una pelota que rebota dentro de un área de dibujo (Canvas).

**Logros:**

* Implementación de concurrencia mediante hilos para animación fluida.
* Creación de una interfaz gráfica interactiva y visual.

## Laboratorio 05: JSP con Apache Tomcat

**Descripción:**  
Se configuró el servidor web Apache Tomcat para desplegar aplicaciones JSP. Se desarrolló y publicó una página JSP que realiza el cálculo de intereses.

**Logros:**

* Configuración exitosa del servidor Apache Tomcat en el entorno local.
* Despliegue y visualización correcta de la página JSP calculoInteres.jsp en el navegador via http://localhost:8090/JSP01/calculoInteres.jsp.

## Laboratorio 06: JavaBean en JSP

**Descripción:**  
Se integró un JavaBean (TourBean) dentro de una página JSP para manejar y mostrar información dinámica, aplicando el patrón MVC básico.

**Logros:**

* Integración funcional de JavaBeans en una aplicación web JSP.
* Uso de métodos get y set para manipulación dinámica de datos.
* Configuración y despliegue adecuado en Apache Tomcat.
* Página accesible vía http://localhost:8090/JSP02/.

## Laboratorio 07: Reserva en JSP

**Descripción:**  
Desarrollo y despliegue de una aplicación web JSP que simula un sistema de reservas, gestionando adecuadamente solicitudes HTTP y rutas.

**Logros:**

* Organización correcta de la estructura del proyecto y configuración del archivo web.xml.
* Evitación de errores comunes como HTTP 404 mediante manejo adecuado de rutas.
* Implementación de una página de bienvenida funcional para mejorar la experiencia del usuario.
* Despliegue exitoso en Apache Tomcat.

## Laboratorio 08: Gestión Académica en Django

**Descripción:**  
Desarrollo de una aplicación web utilizando el framework Django para gestionar entidades académicas como alumnos, cursos y matrículas. Se implementaron modelos relacionales y operaciones de persistencia sobre una base de datos relacional, con interfaz administrativa incorporada.

**Logros:**

* Creación de un proyecto Django estructurado con entorno virtual y configuración inicial correcta.
* Desarrollo del modelo de datos con tres entidades clave: Alumno, Curso y Matriculas, incluyendo relaciones y campos representativos.
* Registro exitoso de la app GestionAcademica en INSTALLED\_APPS y configuración de migraciones.
* Migraciones aplicadas correctamente, creando las tablas en la base de datos.
* Configuración del panel de administración para gestionar alumnos, cursos y matrículas desde una interfaz amigable.
* Creación de un súper usuario para acceso administrativo.
* Despliegue local funcional mediante el servidor de desarrollo de Django.

**Tecnologías utilizadas:**

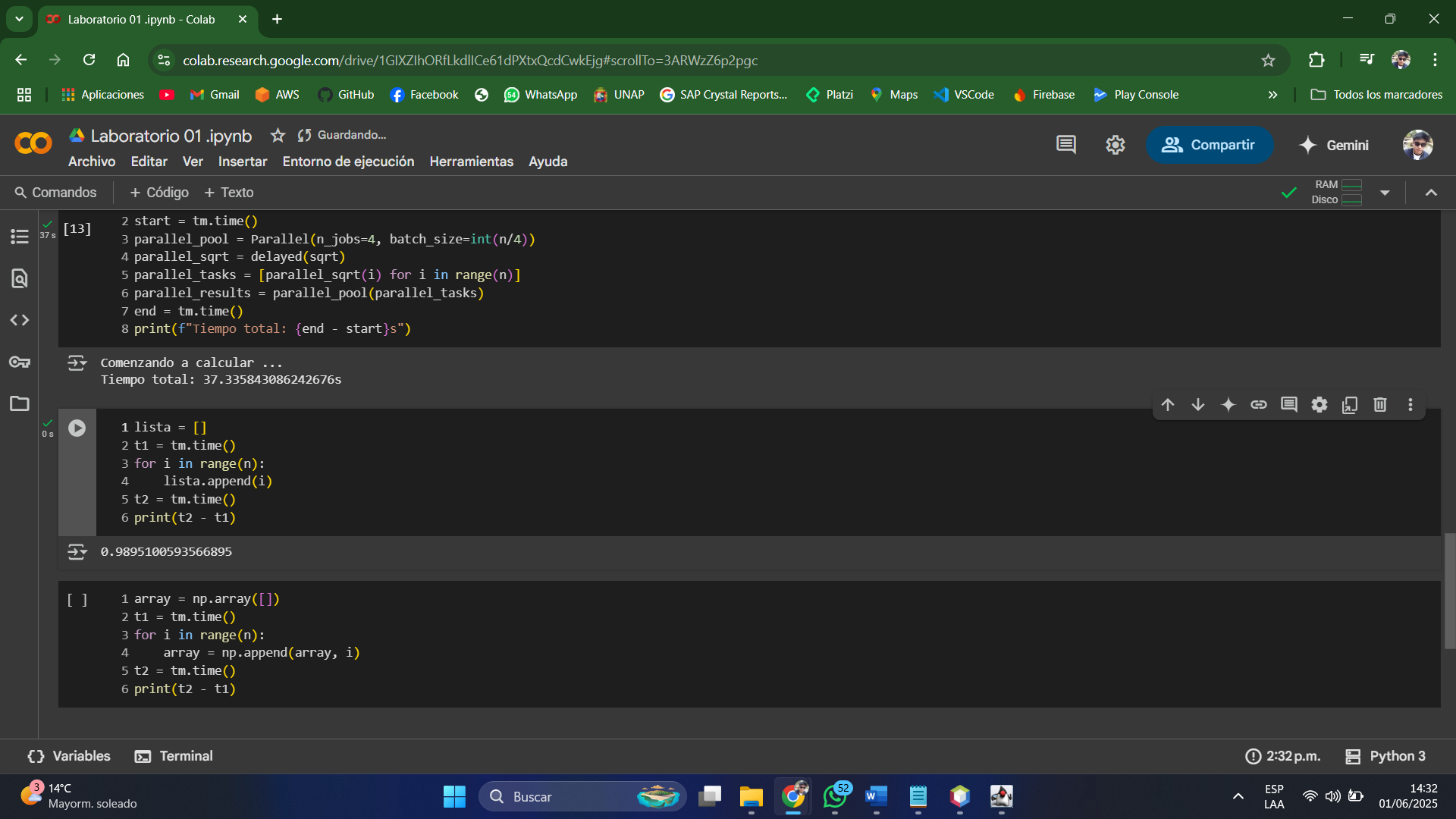
* Python 3
* Django
* SQLite (por defecto)
* Apache Tomcat (solo en laboratorios previos como el 07)
* Entorno virtual (venv)
* Pycharm

## CONSLUSIONES

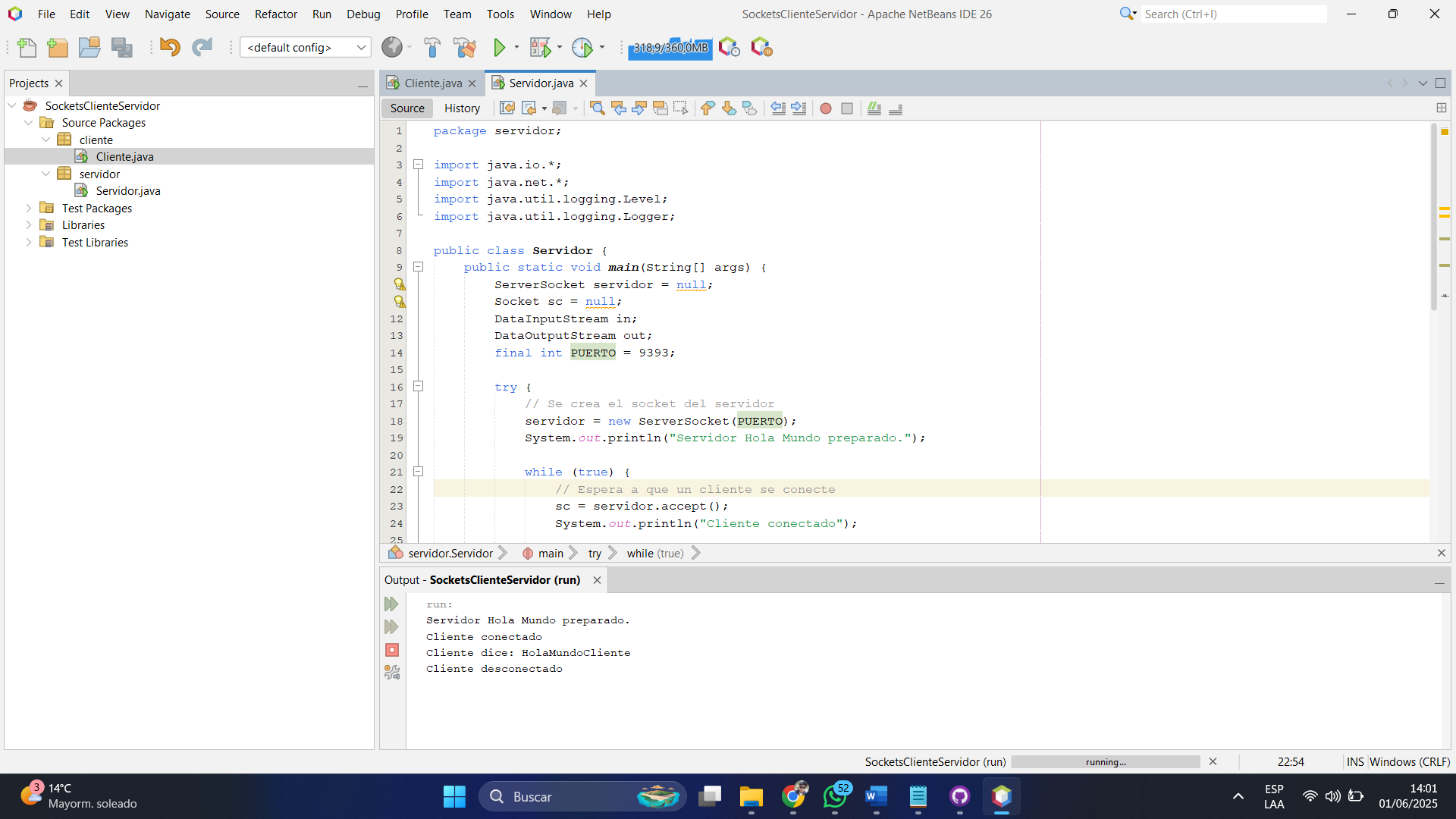
1. **Comprensión Integral de Sistemas Distribuidos.** Los laboratorios realizados permitieron adquirir una visión amplia y práctica sobre los conceptos fundamentales de los sistemas distribuidos, incluyendo la programación concurrente, la comunicación a través de redes y el desarrollo de aplicaciones distribuidas con diferentes tecnologías.
2. **Aplicación de Tecnologías Diversas.** El uso de herramientas y lenguajes variados como Python, Java, JSP, Apache Tomcat y Django facilitó el entendimiento de distintos paradigmas y arquitecturas, desde la programación paralela y concurrente hasta el desarrollo web y la gestión de bases de datos.
3. **Experiencia en Programación Concurrente y Paralelismo.** La comparación de rendimiento entre estructuras de datos nativas y NumPy, así como la paralelización con Joblib, evidenció la importancia de elegir adecuadamente las herramientas y técnicas para optimizar el desempeño en sistemas distribuidos.
4. **Dominio de Comunicación Cliente-Servidor.** La implementación práctica de sockets en Java y el desarrollo con RMI reforzaron el conocimiento sobre protocolos TCP/IP y la invocación de métodos remotos, aspectos esenciales para la comunicación eficiente en aplicaciones distribuidas.
5. **Desarrollo de Aplicaciones Web Distribuidas.** La configuración y despliegue de aplicaciones JSP con Apache Tomcat, así como la integración de JavaBeans y la construcción de sistemas web con Django, demostraron la capacidad para crear soluciones distribuidas con interfaces interactivas y gestión dinámica de datos.
6. **Fortalecimiento de Habilidades Prácticas.** El desarrollo de proyectos completos, desde la animación con hilos en Java hasta la gestión académica con Django, contribuyó al fortalecimiento de habilidades técnicas, resolución de problemas y buenas prácticas de programación en entornos distribuidos.
7. **Preparación para Entornos Reales.** Los ejercicios y configuraciones realizadas en los laboratorios simulan escenarios reales de sistemas distribuidos, lo que prepara al alumno para enfrentar desafíos de diseño, implementación y despliegue de aplicaciones distribuidas en entornos profesionales.

# CAPTURAS

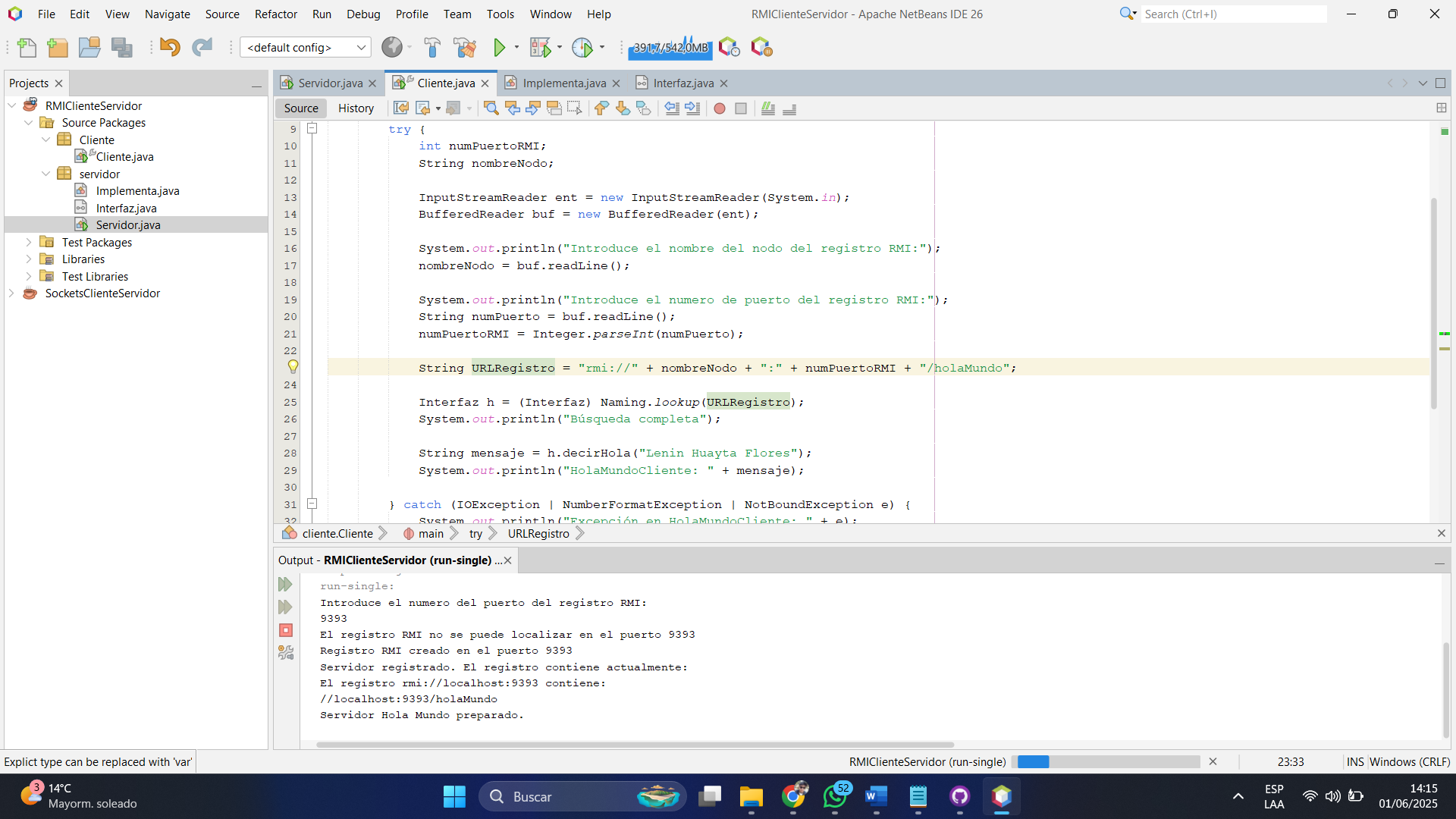
## LABORATORIO 01

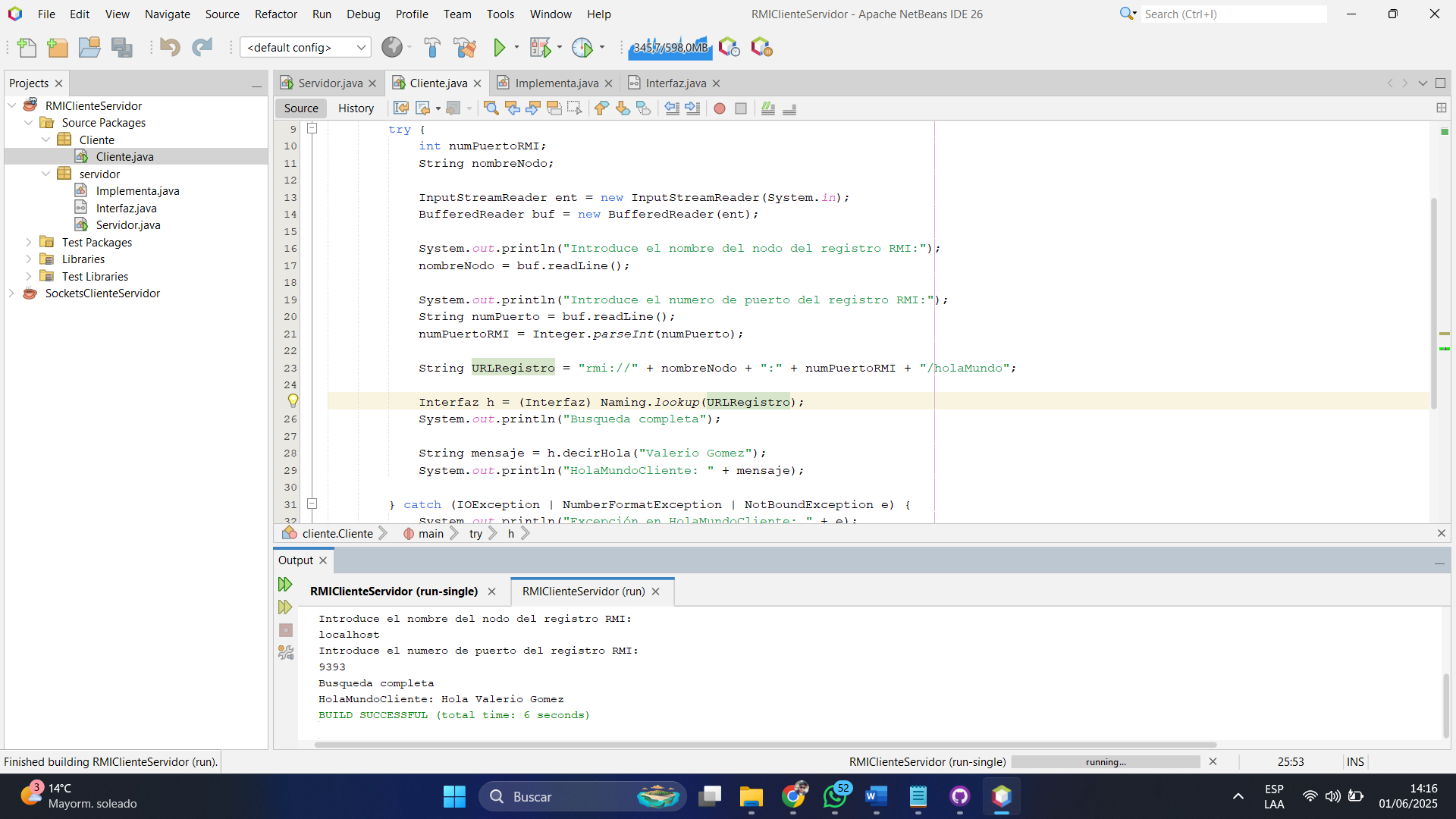


## LABORATORIO 02

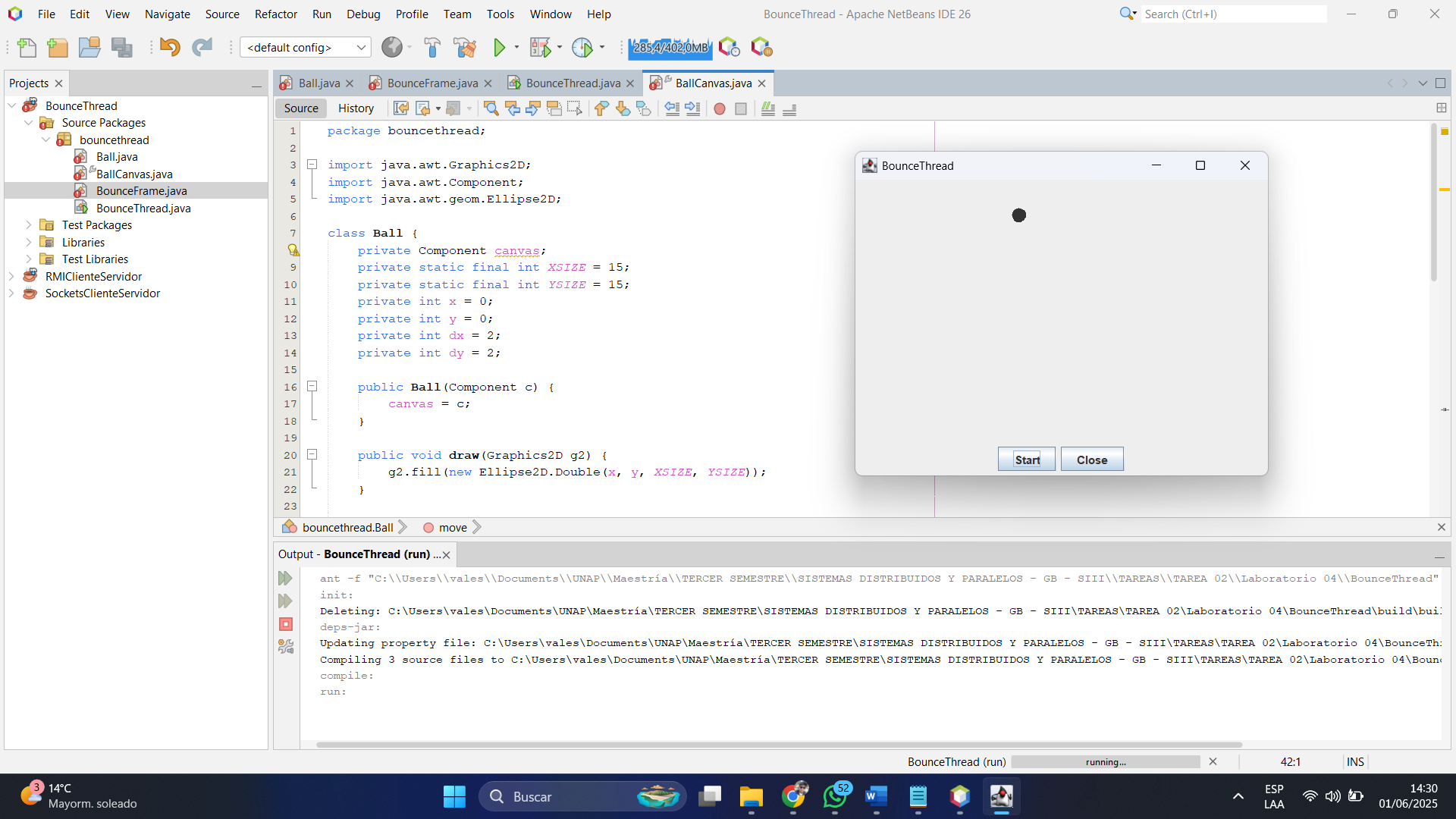


## LABORATORIO 03

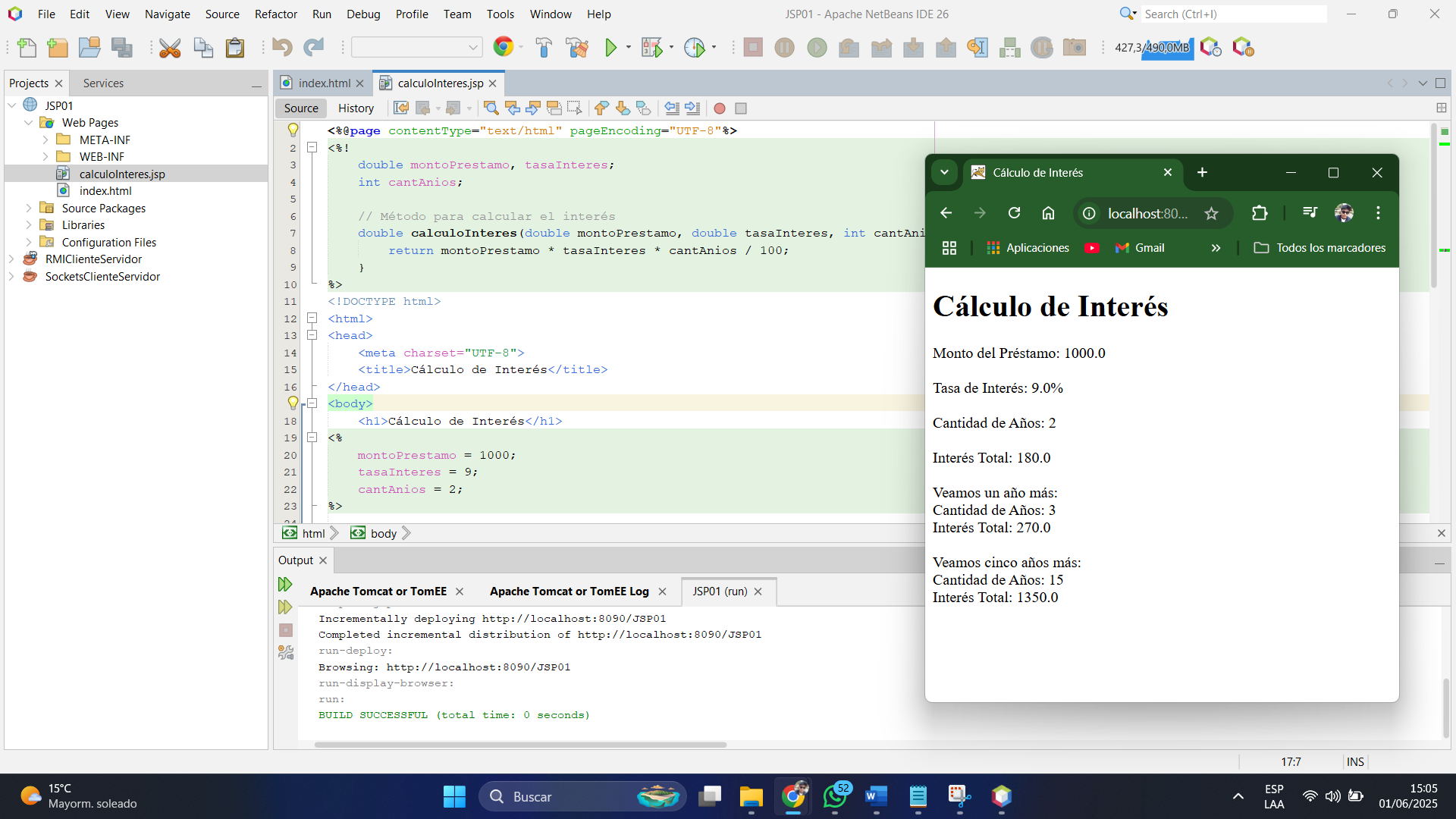




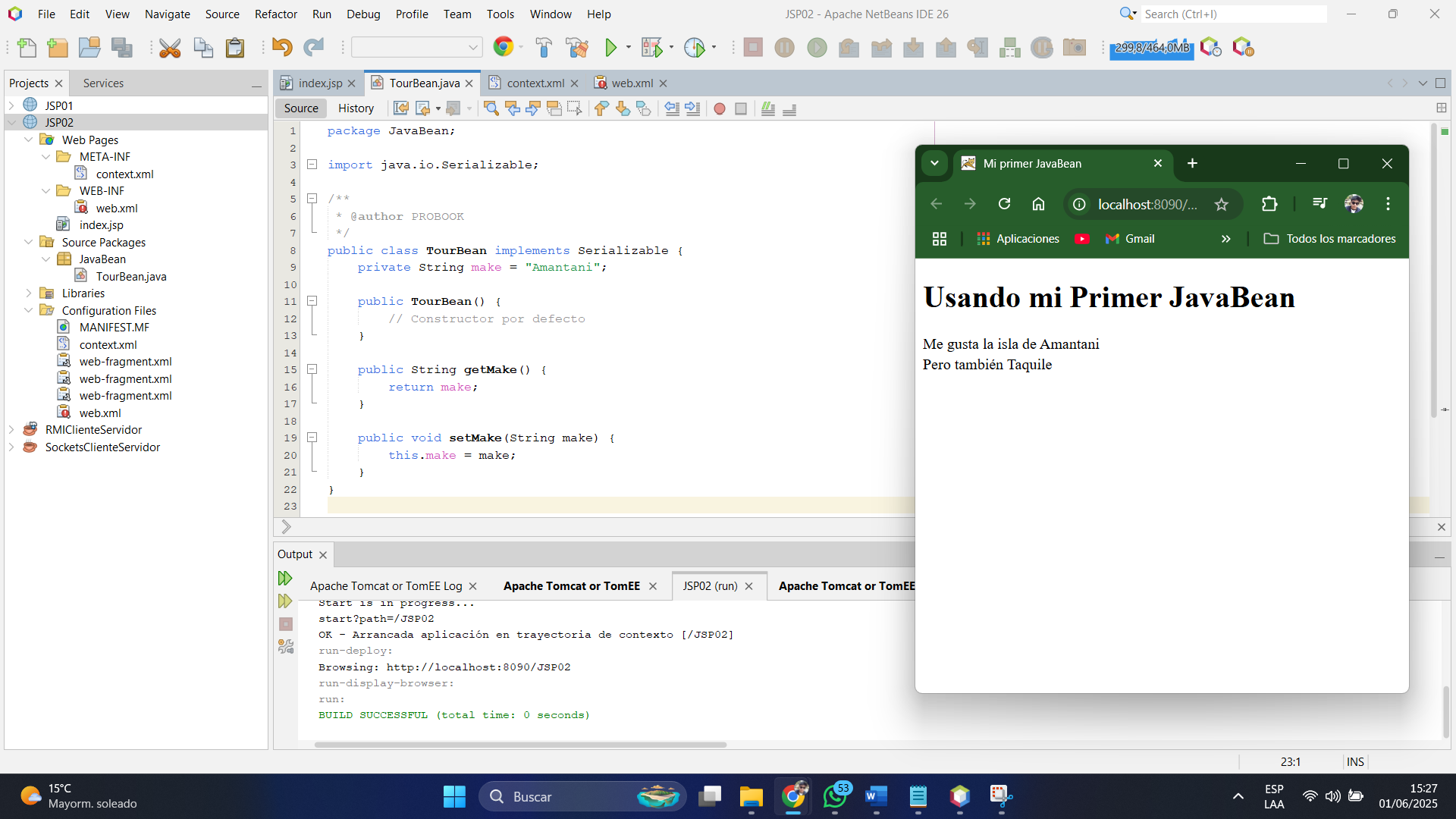
## LABORATORIO 04



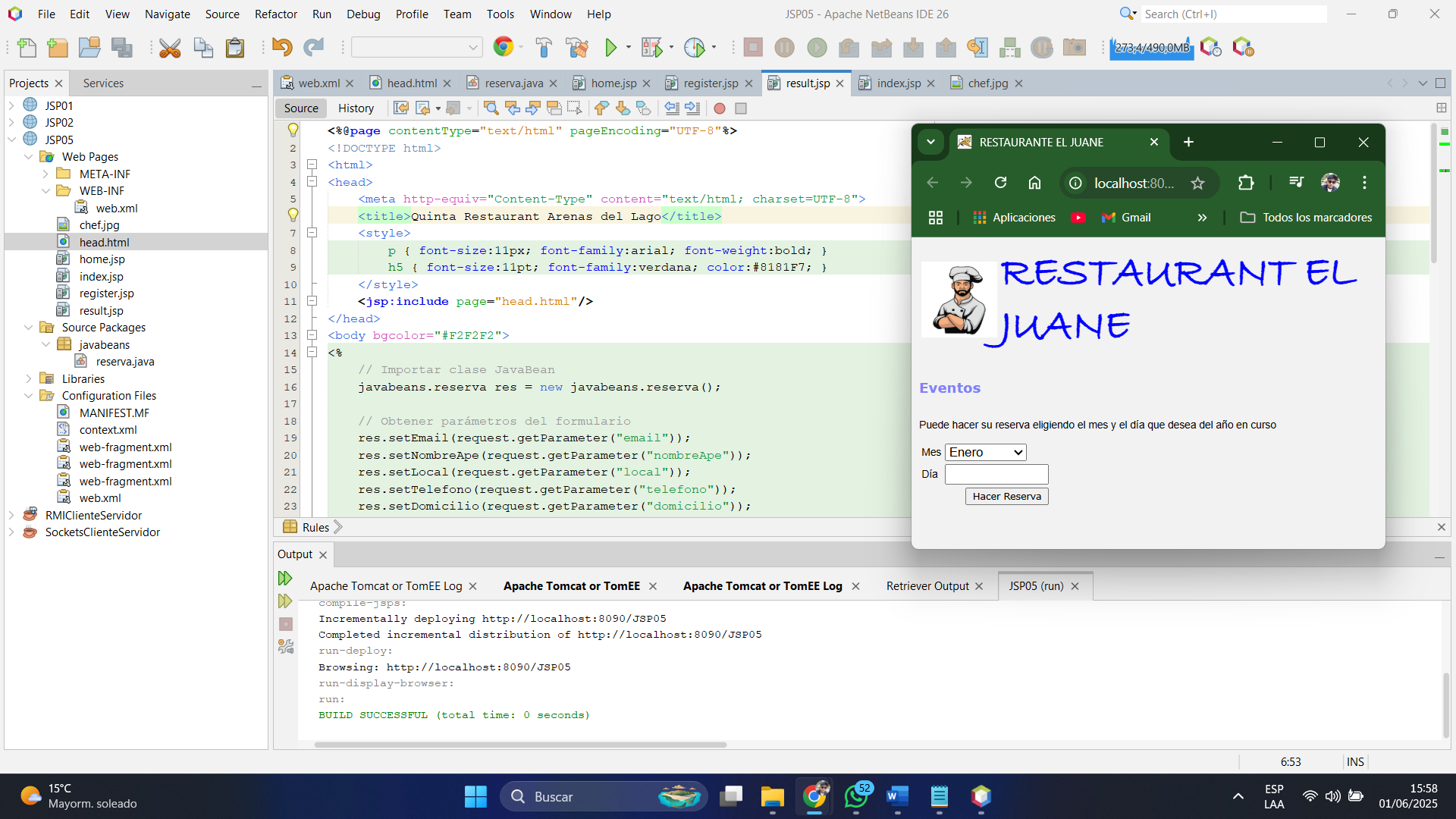
## LABORATORIO 05

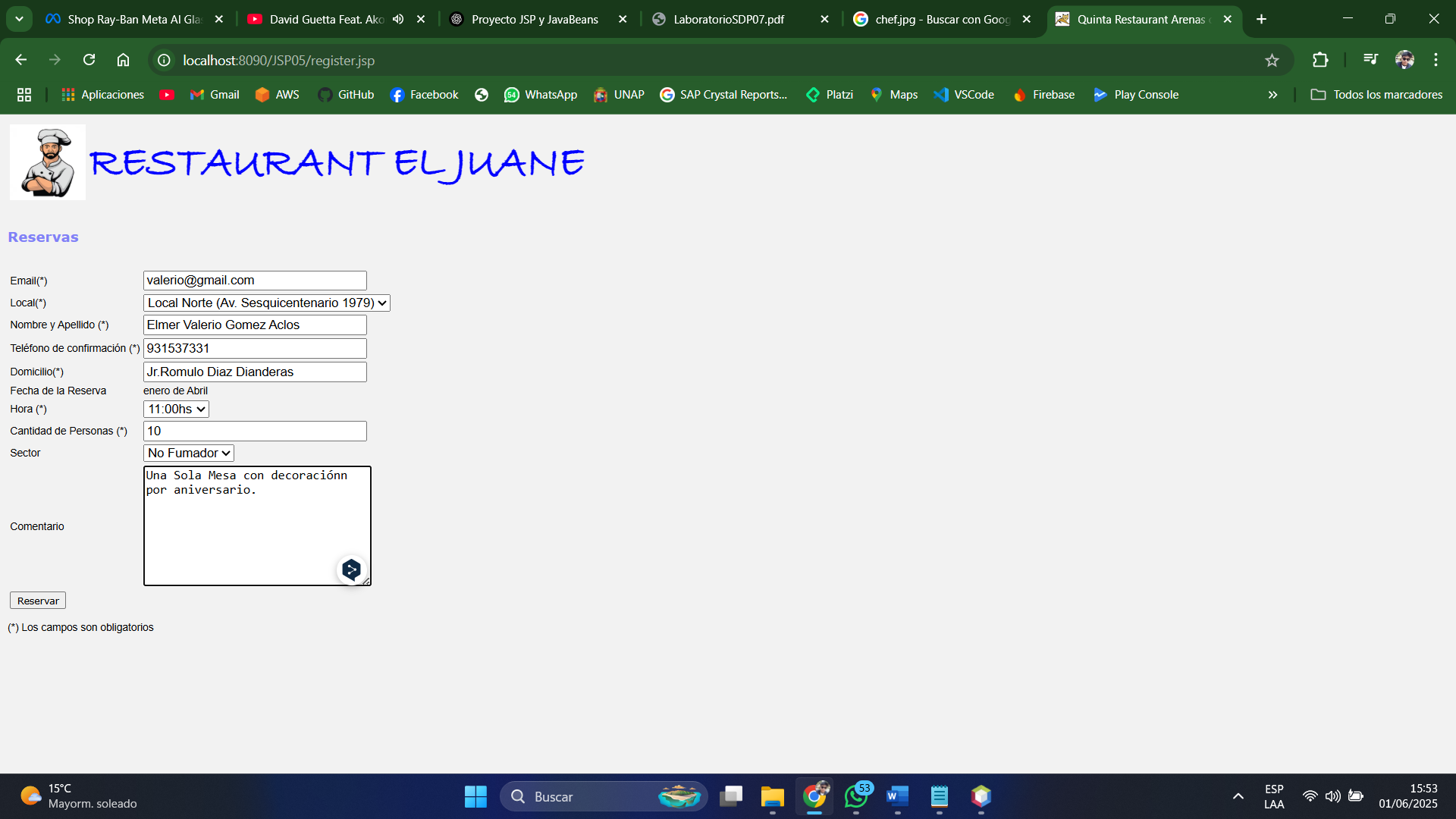
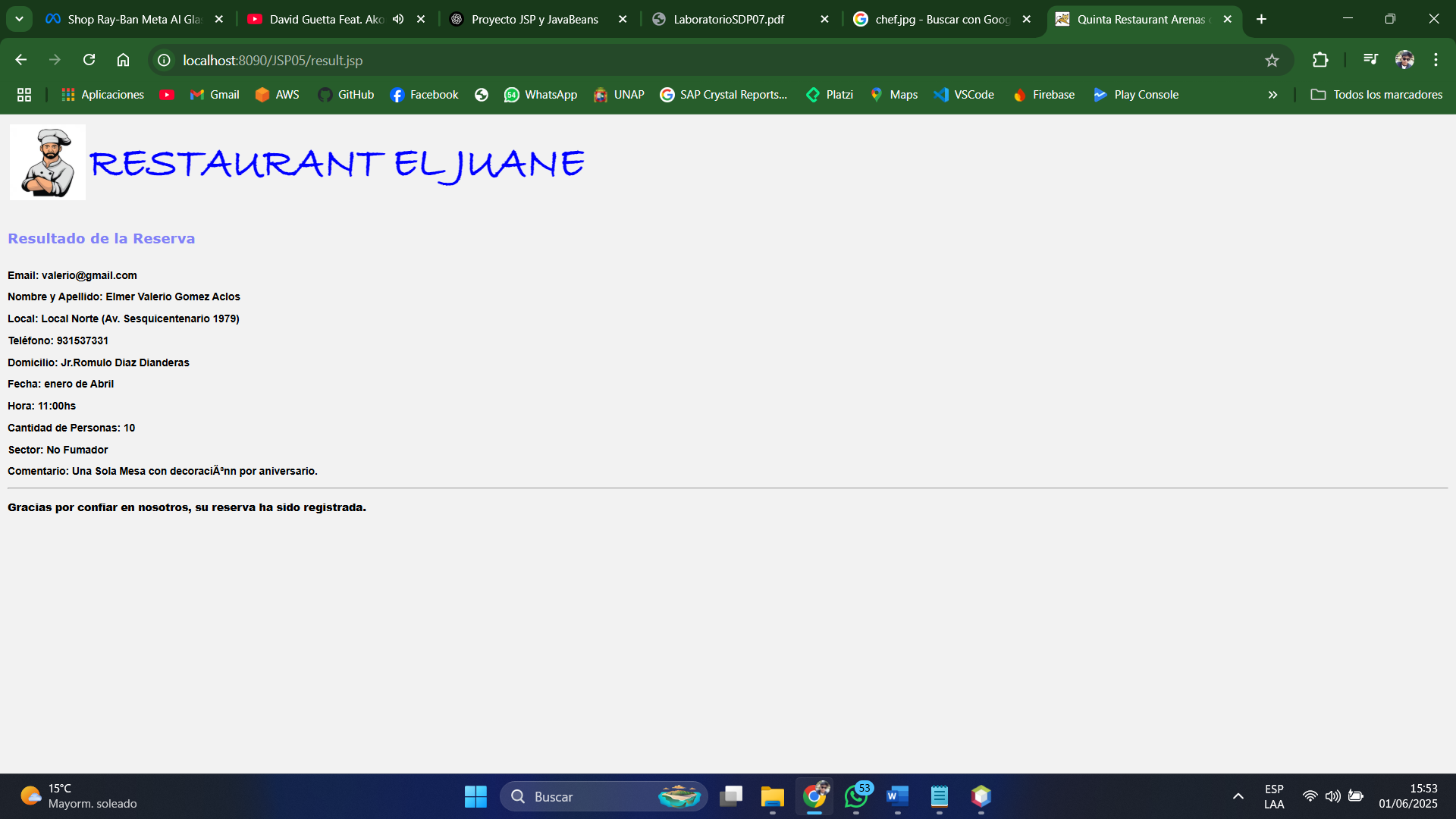


## LABORATORIO 06



## LABORATORIO 07





## LABORATORIO 08

