

# Segundo Proyecto de Tercio

Juan Camilo Mantilla Rubio, Daniel Ospina Bedoya,  
Sebastián Camilo Reyes Villamil

**Abstract**—Este es un artículo que describe los pasos realizados y los temas que se tocaron para la realización de este proyecto. La intención de este proyecto era crear un servidor REST usando Spring, que reciba un GET con parámetro numérico y retorne el cuadrado del número y desplegarlo en Heroku. Luego se crea una máquina virtual Linux en AWS y se extiende el servidor web que se construyó para que funcione como servidor de aplicaciones usando el patrón de inversión de control. A continuación, se crea una aplicación WEB, usando el servidor, que tenga un formulario que pida al usuario un número y le regrese el cuadrado de este. Esta aplicación se despliega en AWS. Al final se hacen unas pruebas para comprobar que todo quedó bien y funciona correctamente. Una vez comprobado su funcionamiento se sube a un repositorio en GitHub para que lo puedan ver y utilizar las personas interesadas.

**Index Terms**—Amazon Web Services(AWS), GitHub, Heroku, servidor REST, Spring.

## I. INTRODUCCIÓN

ESTE documento es una descripción del proyecto que realizamos. El proyecto consistía en montar un API REST en Heroku que calculara el cuadrado de un número y devolviera una cadena JSON con el número que ingresamos y el valor de su cuadrado. Luego se montaba un servidor web en una máquina virtual de Amazon Web Services, que mostrará una página web con un formulario, en la cual el usuario pudiera ingresar el número y le muestre el valor del número al cuadrado. Para esto el servidor web le envía al API REST que está en Heroku el número que el usuario ingreso para que este haga el cálculo y le envíe la respuesta al servidor web para que este se la muestre al usuario.

## II. PASOS PARA LA REALIZACIÓN DEL PROYECTO

Para la realización de este proyecto se realizaron diferentes pasos, estos pasos fueron:

### A. Creación del API REST

En Java, se crea un API REST con Spring que reciba una solicitud HTTP GET con parámetro el número al se le quiere hallar el cuadrado, el cual crea un objeto Response que contiene el número ingresado y su cuadrado. Esta clase retorna una cadena JSON que contiene el valor ingresado y su cuadrado. Luego, este API se despliega en Heroku.

### B. Creación del servidor web

Se crea un servidor en Java que busca una variable de ambiente para el puerto que utilizará, si no la encuentra, correrá en el puerto 8081. Luego, este servidor crea un Socket y una Thread Pool que permitirá el uso concurrente de la aplicación, el servidor lanzará Thread Servers que son los que atenderán las solicitudes de los usuarios.

### C. Creación máquina virtual AWS

Se crea en AWS una máquina virtual de Linux, a esta máquina se suben las clases compiladas en Java 1.7 usando sftp y se pone en ejecución. Una vez montado todo, se ingresa a la URL de Heroku para confirmar que está desplegado el API, después se accede a la IP de la máquina virtual de Linux en AWS, por el puerto 8081, donde se muestra la página con el formulario para ingresar el número y luego nos devuelve la respuesta.

## III. ARQUITECTURA DEL PROYECTO

El patrón que se utilizó en este proyecto fue de cliente servidor, en el cual un usuario que quiere saber el cuadrado de un número se conecta por medio de internet a un servidor web en una máquina virtual de Linux que está montada en AWS. Este servidor web se conecta a la aplicación web que está subida en un micro contenedor de Heroku para que haga los cálculos y le retorne al servidor web la respuesta. Una vez el servidor web recibe la respuesta, la muestra al usuario. Estas comunicaciones entre los servidores se hacen por internet por medio del protocolo HTTP.

## IV. CONCLUSIONES

Con la realización de este proyecto pudimos concluir que las plataformas de Heroku y AWS son muy útiles a la hora de desplegar servicios en línea.

Aprendimos a construir una arquitectura que utiliza diferentes servicios y capas para cada parte de su funcionamiento.

También, nos pudimos dar cuenta que el patrón de cliente – servidor es conveniente a la hora de desplegar páginas sencillas y permite que se tenga un muy buen rendimiento en el servicio.

Por último, nos pareció que las herramientas de AWS facilitan mucho el trabajo, ya que pone a disposición de los desarrolladores muchas funciones que serían muy complicadas de hacer todas desde cero y que necesitan de un mantenimiento constante, que ya nos proporciona AWS.