A low-angle, upward-looking photograph of several modern skyscrapers with glass facades. The buildings are reflected in each other, creating a complex geometric pattern. The sky is visible through the glass panels, showing a blue color with some white clouds. The perspective makes the buildings appear to converge towards the top of the frame.

# Lenguaje de Programación II

Fundamentos de la Programación Web

Richard Bardales

# Capítulo 2

## Fundamentos WEB

**01** Conocer los conceptos generales de la web.

**02** Identificar los lenguajes para crear aplicaciones web.

**03** Crear una aplicación web básica.



# Capítulo 2



## Conceptos generales de la web

Lo que hoy se conoce como la Web, es parte de la WWW, que en su definición anglosajona quiere decir World Wide Web, lo que en español significa Red Mundial de contenido, este sistema de comunicación, utiliza un medio para su propósito de comunicación, que está formado por un conjunto descentralizado de redes interconectadas, que a su vez se basan en la familia de protocolos TCP/IP para garantizar un estandar, sin importar que tan heterogeneas sean las estructuras físicas de redes implementadas en diferentes partes del mundo.

La web como se le conoce comunmente permite compartir contenido a través del protocolo de Transferencia de Hipertexto (HTTP), y su característica principal es la navegación a través de enlaces a diversas páginas web, para la creación de estas páginas web se utiliza el lenguaje de marcado de hipertexto (HTML), para hacer el enlace a las diferentes páginas utiliza un sistema de Localización Uniforme de Recursos (URL), que es un identificador único para páginas, documentos, archivos u otros recursos que se encuentran en la web.



## Historia

En 1989 Tim Berners-Lee quien trabajaba para CERN fue el creador del primer servidor web, el primer navegador web y un protocolo para dar formato a los documentos de la web, denominado Lenguaje de Marcado de Hipertexto (HTML), tomando como referencia los conceptos de hipervínculos que ya existía anteriormente, la publicación del HTML

se hizo en 1991 y el código fuente del navegador web fue liberado en 1993, con esto se crearon muchos mas navegadores web y uno de ellos fue Mosaic, que luego pasó a llamarse Netscape, debido a su facilidad de uso sin la necesidad de tener muchos conocimientos técnicos, fue quien desencadenó la popularidad del internet en la década de los 90

## Funcionamiento de la web

Cuando hacemos click en un enlace de nuestro explorador web, el primer paso es identificar en todo el URL, el nombre de dominio y traducir ese URL en su IP asociada que se encuentra en la base de datos, en los Servidores de Nombres de Dominio (DNS), una vez que se ubica la dirección IP del servidor web, se puede hacer el envío de paquetes de datos.

El paso número dos consiste en enviar peticiones bajo el protocolo HTTP a nuestro servidor web, solicitando el recurso específico (página web), inicialmente se envía el Hipertexto y luego los demás archivos multimedia y otros asociados a la página web.

El paso número tres consiste en que el navegador web recibe los archivos solicitados desde el servidor web y los renderiza (carga las etiquetas html, los CSS, los eventos y funciones con JavaScript), finalmente el usuario ve una pantalla con el contenido solicitado.

## Estándares web

Destacamos los siguientes estándares:

**El Identificador de Recurso Uniforme (URI)**, que es un sistema universal para referenciar recursos en la Web, como páginas web

**El Protocolo de Transferencia de Hipertexto (HTTP)**, que especifica cómo se comunican el navegador y el servidor entre ellos

**El Lenguaje de Marcado de Hipertexto (HTML)**, usado para definir la estructura y contenido de documentos de hipertexto

**El Lenguaje de Marcado Extensible (XML)**, usado para describir la estructura de los documentos de texto.

Berners-Lee dirige desde 2007 el World Wide Web Consortium (W3C), el cual desarrolla y mantiene esos y otros estándares que permiten a los ordenadores de la Web almacenar y comunicar efectivamente diferentes formas de información.

## Servidor Web

Un servidor web (server) es un ordenador de gran potencia que se encarga de “prestar el servicio” de transmitir la información pedida por sus clientes (otros ordenadores, dispositivos móviles, impresoras, personas, etc.)

Los servidores web (web server) son un componente de los servidores que tienen como principal función almacenar, en web hosting, todos los archivos propios de una página web (imágenes, textos, videos, etc.) y transmitirlos a los usuarios a través de los navegadores mediante el protocolo HTTP (Hypertext Transfer Protocol).

El rol principal de un servidor web es almacenar y transmitir el contenido solicitado de un sitio web al navegador del usuario.

Este proceso, para los internautas no dura más que un segundo, sin embargo, a nivel del web server es una secuencia más complicada de lo que parece.

Para cumplir con sus funciones el servidor deberá tener la capacidad de estar siempre encendido para evitar interrumpir el servicio que le ofrece a sus clientes. Si dicho servidor falla o se apaga, los internautas tendrán problemas al ingresar al sitio web.

La comunicación entre un servidor y sus clientes se basa en HTTP, es decir, en el protocolo de transferencia de hipertexto o en su variante codificada HTTPS.

Para saber cómo funciona, primero es necesario conocer que el web server está permanentemente en espera de una solicitud de información.

Además, ten en cuenta que toda computadora, smartphone o tablet tiene una dirección IP única e irrepetible que lo identifica de otro dispositivo en la red, así es como el servidor web envía la información exacta que el internauta está esperando.

Ahora bien, para que el web server pueda cumplir con su función es necesario que reciba la petición por parte de un navegador, en otras palabras, se envía un pedido desde una dirección IP hacia la dirección IP del servidor que aloja los archivos del sitio en cuestión.

A continuación, el servidor web busca en sus archivos la información que se le está solicitando, procede a interpretar las líneas de código y a enviar el resultado al navegador cuya dirección IP fue la solicitante.

Este resultado se le muestra a los internautas y es lo que siempre sucede cuando se navega en sitios de Internet. Cuando este proceso se completa podemos decir que el web server ha cumplido con su función.

## **Tipos de Servidores Web**

### **Apache**

Es el más común y utilizado en el mundo, sin embargo, ha perdido popularidad frente a Microsoft IIS y Nginx.

Entre las ventajas de Apache está que es un código abierto, con software gratuito y multiplataforma, y entre sus desventajas su bajo rendimiento cuando recibe miles de requests (peticiones) simultáneas en procesamiento de contenido dinámico o archivos estáticos.

### **Nginx**

Conocido y popularizado como una de las mejores alternativas de Apache. Nginx es un servidor web de código abierto y gratuito (aunque también existe una versión comercial) que se destaca por su alto rendimiento.

Entre sus beneficios resalta una configuración simple, ligera, rápida y excelente en cuanto a seguridad y rendimiento, además permite ser configurado para integrarse nativamente con casi cualquier tecnología y lenguaje de programación moderno.

Como desventajas podemos encontrar que no soporta los archivos .htaccess (de Apache), aunque incluye su propio lenguaje de rewrites.

## **Tomcat**

Apache Tomcat (también llamado Jakarta Tomcat o simplemente Tomcat) funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las especificaciones de los servlets y de JavaServer Pages (JSP) de Oracle Corporation (aunque creado por Sun Microsystems).

## **Microsoft IIS**

Internet Information Services o IIS se ha popularizado para ofrecer servicios en la nube, principalmente en Azure (la plataforma de Cloud Hosting de Microsoft).

Además, su perfecta integración con Windows (claro está), Visual Studio y sus herramientas hicieron que este web server se posicionara como el servidor líder en el mundo empresarial.

# **HTML**

HTML (Lenguaje de Marcas de Hipertexto, del inglés HyperText Markup Language ) es el componente más básico de la Web. Definir el significado y la estructura del contenido web. "Hipertexto" hace referencia a los enlaces que conectan páginas web entre sí, ya sea dentro de un único sitio web o entre sitios web. Los enlaces son un aspecto fundamental de la Web. Al subir contenido a Internet y vincularlo a las páginas creadas por otras personas, te conviertes en un participante activo en la « World Wide Web » (Red Informática Mundial).

# **Java Script**

JavaScript, en cambio, es un lenguaje de script que inicialmente fue desarrollado para ser usado dentro de las páginas web. La versión estandarizada es el ECMAScript. Si bien los nombres son similares, JavaScript fue desarrollado por Netscape y no tiene relación alguna con Java, aparte de que sus sintaxis derivan del lenguaje de programación C. En unión con el Document Object Model de una página web, JavaScript se ha convertido en una tecnología mucho más importante de lo que pensaron sus creadores originales. La manipulación del Modelo de Objetos de Documento después de que la página ha sido

enviada al cliente se ha denominado HTML Dinámico (DHTML), para enfatizar un cambio con respecto a las visualizaciones de HTML estático.

En su forma más simple, toda la información opcional y las acciones disponibles en las páginas web con JavaScript ya son cargadas la primera vez que se envía la página. Ajax ("Asynchronous JavaScript And XML", en español, JavaScript Asíncrono y XML) es una tecnología basada en JavaScript que puede tener un efecto significativo para el desarrollo de la Web. Ajax proporciona un método por el cual grandes o pequeñas partes dentro de una página web pueden actualizarse, usando nueva información obtenida de la red en respuesta a las acciones del usuario. Esto permite que la página sea mucho más confiable, interactiva e interesante, sin que el usuario tenga que esperar a que se cargue toda la página. Ajax es visto como un aspecto importante de lo que suele llamarse Web 2.0. Ejemplos de técnicas Ajax usadas actualmente pueden verse en Gmail, Google Maps, etc.

## CSS

CSS es uno de los lenguajes más importantes que se utilizan para ordenar las instrucciones referentes a la apariencia de un sitio y presentar los contenidos de una página de forma atractiva. De este modo, HTML se emplea para estructurar el contenido de un sitio, mientras que CSS, para estructurar su presentación.

## SASS

SASS es un preprocesador de CSS compatible con todas sus versiones. Por lo tanto, se trata de una herramienta utilizada por los desarrolladores web para traducir un código de hojas de estilo no estándar a un código CSS estándar, legible por la mayoría de los navegadores.

## BOOTSTRAP

Bootstrap es un marco de desarrollo front-end gratuito y de código abierto para la creación de sitios web y aplicaciones web. Diseñado para permitir el desarrollo responsivo de sitios web móviles , Bootstrap proporciona una colección de sintaxis para diseños de plantillas.

El marco Bootstrap se basa en lenguaje de marcado de hipertexto ( HTML ) , hojas de estilos en cascada ( CSS ) y JavaScript . Los desarrolladores web que utilizan Bootstrap pueden crear sitios web mucho más rápido sin perder tiempo preocupándose por comandos y funciones básicas.

# RESPONSIVE

Cuando utilizamos el término “responsive” (adaptable), nos referimos principalmente a “responsive design” (diseño web adaptable). Esto significa hacer que un sitio web sea accesible y adaptable en todos los devices: tabletas, smartphones, etc..

# FRAMEWORK

Un framework es un esquema o marco de trabajo que ofrece una estructura base para elaborar un proyecto con objetivos específicos, una especie de plantilla que sirve como punto de partida para la organización y desarrollo de software.

# JSON

JSON (JavaScript Object Notation) es un formato ligero de intercambio de datos. JSON es de fácil lectura y escritura para los usuarios. JSON es fácil de analizar y generar por parte de las máquinas. JSON se basa en un subconjunto del lenguaje de programación JavaScript, Estándar ECMA-262 3a Edición - Diciembre de 1999..

# GESTOR DE CONTENIDO

Un gestor de contenidos o CMS (del inglés, Content Management System) es una aplicación web o software que permite crear, administrar y publicar una página web en Internet sin necesidad de tener conocimientos de programación.

Estas aplicaciones tienen un panel de administración, también llamada «interfaz», que te permite crear, editar o publicar contenido web.

Para que la creación y gestión de las páginas web sea todavía más sencilla, los gestores de contenido normalmente poseen plugins, extensiones o complementos que permiten añadir o modificar funcionalidades en tu sitio.

# GIT



Git es un proyecto de código abierto maduro y con un mantenimiento activo que desarrolló originalmente Linus Torvalds, el famoso creador del kernel del sistema operativo Linux, en 2005. Un asombroso número de proyectos de software dependen de Git para el control de versiones, incluidos proyectos comerciales y de código abierto. Los desarrolladores que han trabajado con Git cuentan con una buena representación en la base de talentos disponibles para el desarrollo de software, y este sistema funciona a la perfección en una amplia variedad de sistemas operativos e IDE (entornos de desarrollo integrados).

Git, que presenta una arquitectura distribuida, es un ejemplo de DVCS (sistema de control de versiones distribuido, por sus siglas en inglés). En lugar de tener un único espacio para todo el historial de versiones del software, como sucede de manera habitual en los sistemas de control de versiones antaño populares, como CVS o Subversion (también conocido como SVN), en Git, la copia de trabajo del código de cada desarrollador es también un repositorio que puede albergar el historial completo de todos los cambios.

Además de contar con una arquitectura distribuida, Git se ha diseñado teniendo en cuenta el rendimiento, la seguridad y la flexibilidad.

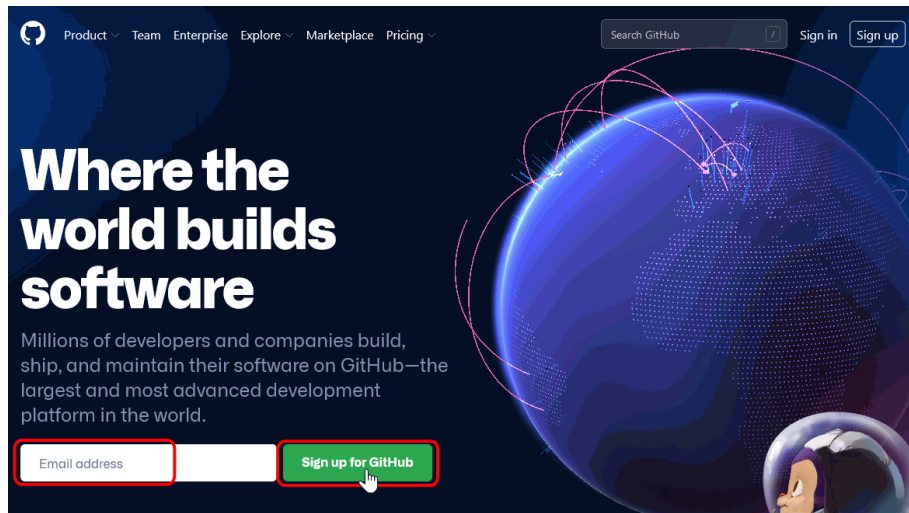
---

## Crear una cuenta en GITHUB

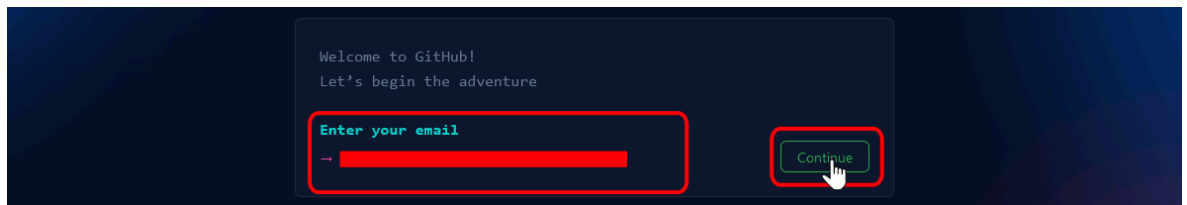
---

Crear una cuenta de GitHub es muy sencillo. Tan sólo se necesita disponer previamente de una cuenta de correo electrónico (gmail, etc.):

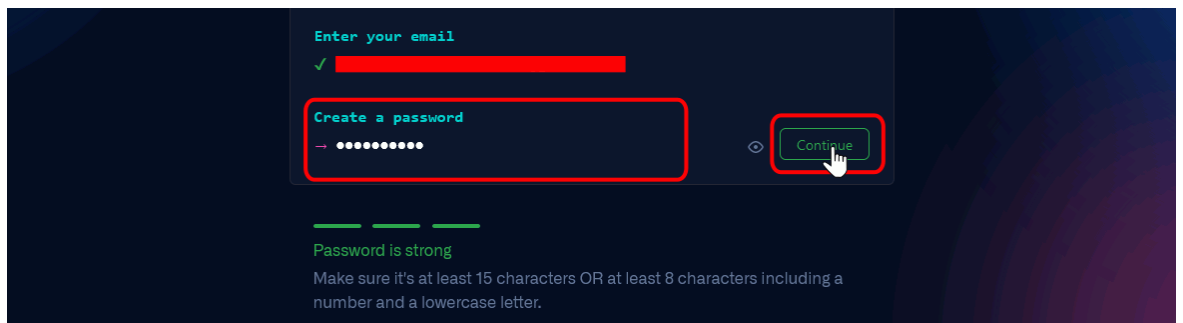
1. Abra la página <https://github.com>.
2. Escriba la dirección de correo con la que quiera vincular su cuenta de GitHub:



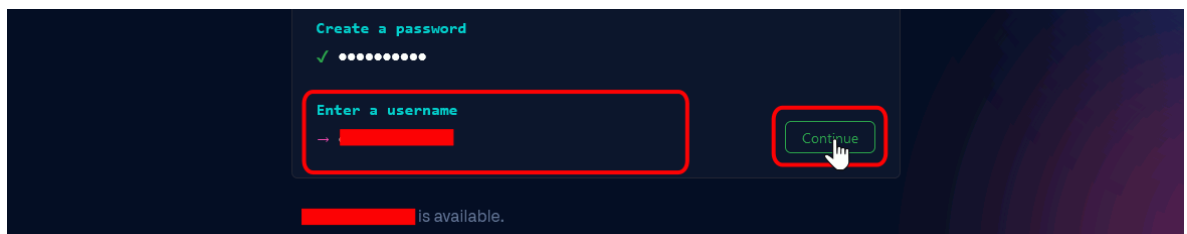
3. Confirme la dirección de correo:



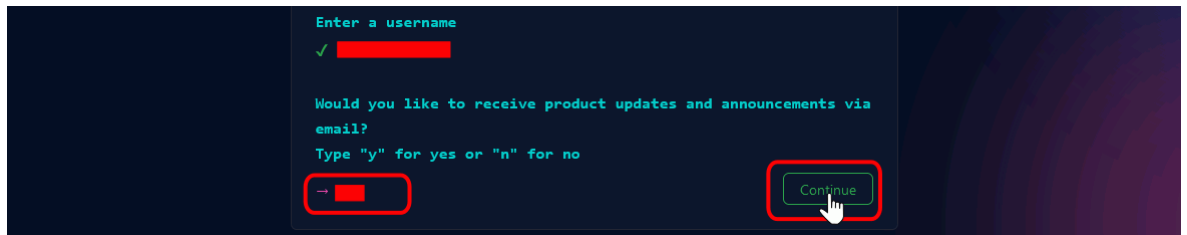
4. Escriba la contraseña para su cuenta de GitHub:



5. Escriba su nombre de usuario de GitHub:



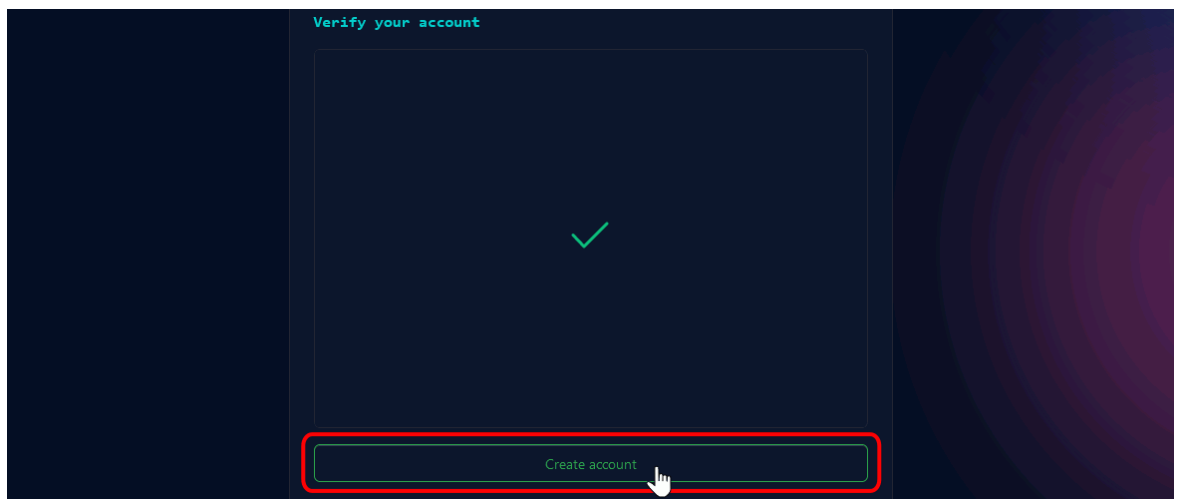
6. Indique si quiere recibir información sobre actualizaciones y nuevos productos de GitHub:



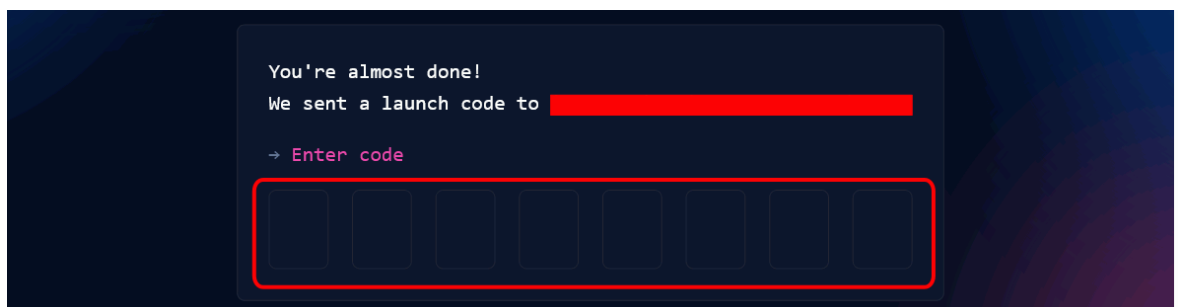
7. Verifique que es una persona resolviendo un sencillo "rompecabezas" (tipo *captcha*):



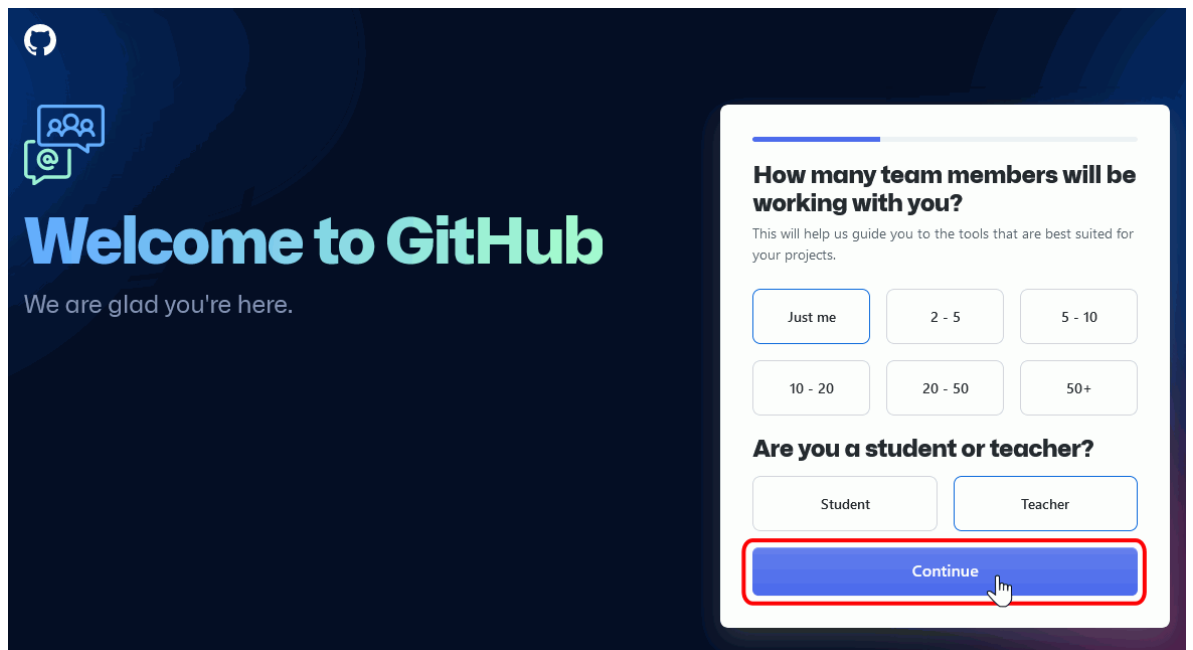
8. Indique que quiere crear la cuenta:



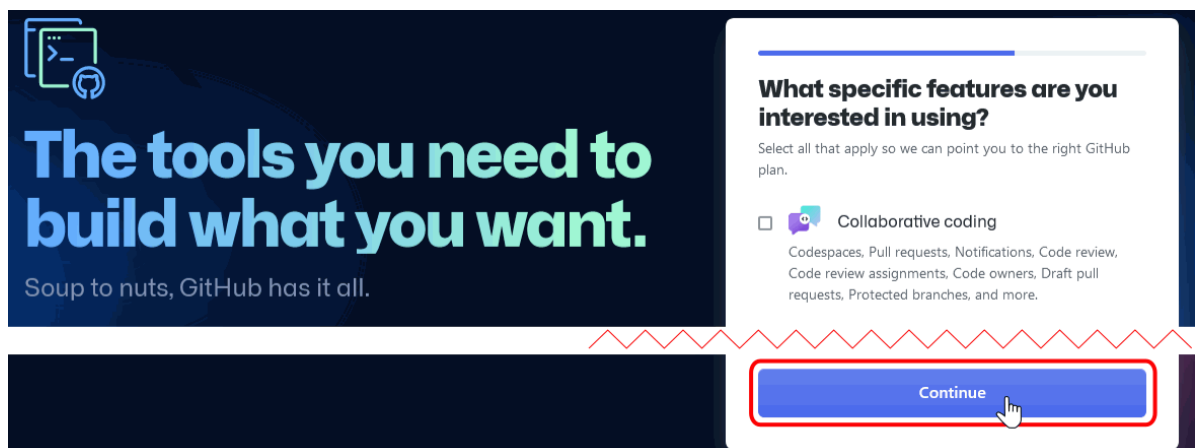
9. GitHub enviará a su cuenta de correo un código de verificación. Escriba el código recibido:



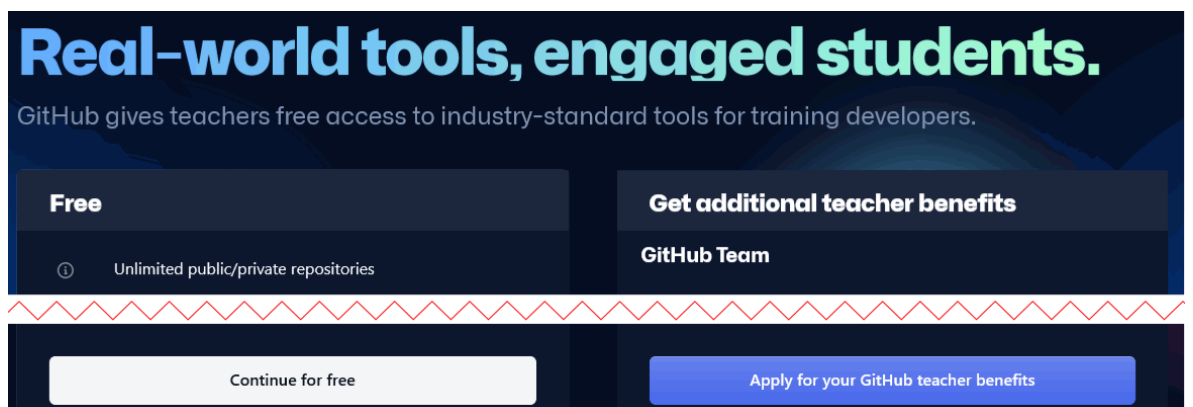
10. Para ayudar a la configuración inicial de la cuenta, se deben responder unas preguntas
11. Indique si es estudiante o profesor y el número de individuos con los que va a trabajar en GitHub:



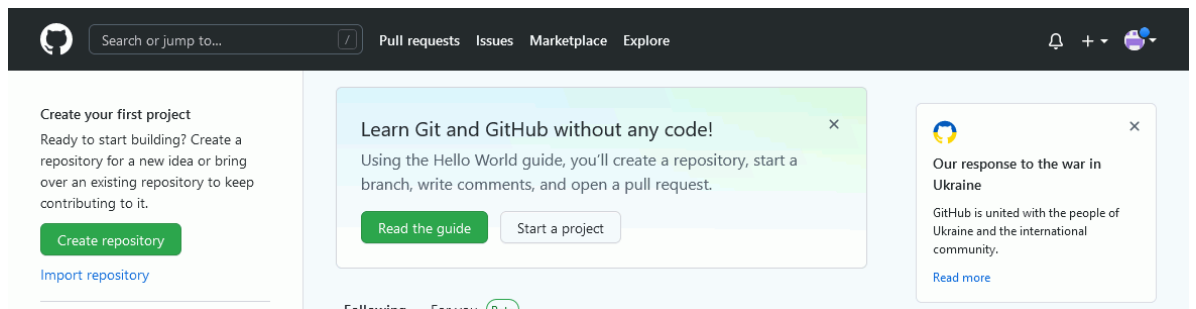
12. Indique si va a utilizar determinadas funcionalidades de GitHub:



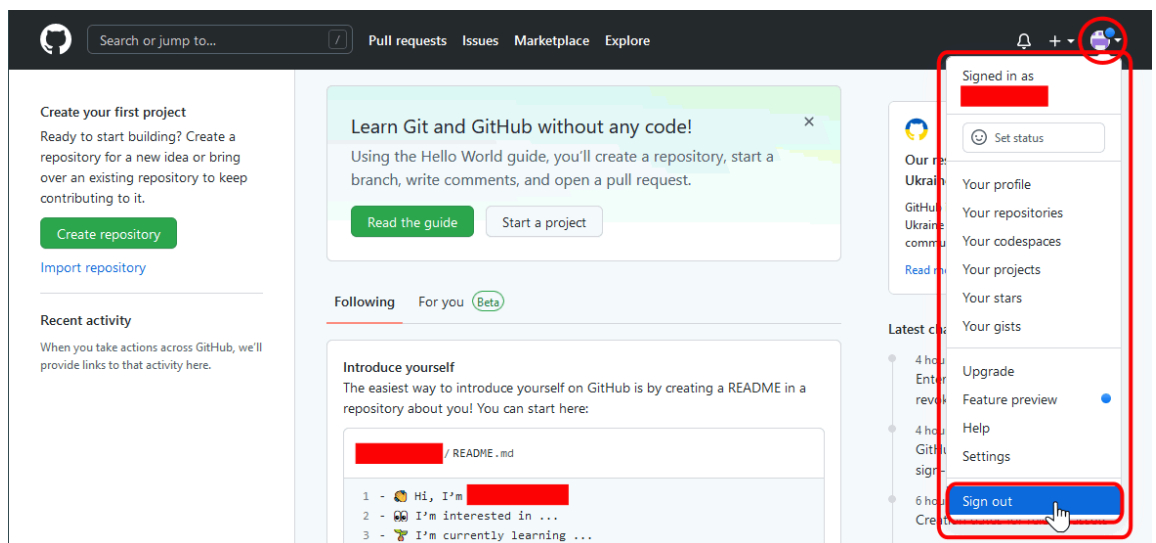
13. Indique si va a solicitar las funcionalidades puestas a disposición de profesores:



14. Inmediatamente se mostrará su panel de usuario de GitHub:




15. Para salir de GitHub, haga clic en el icono superior derecho y elija la opción "Sign out":



## Configuración del GITUB

Antes que nada, tenemos que descargar e instalar Git

Cerca de 930,000,000 resultados (0.48 segundos)

 [git-scm.com](https://git-scm.com)  
<https://git-scm.com> · Traducir esta página

**Git**

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency. Visitaste esta página varias veces. Última visita: 6/12/22.

**Downloads**  
Windows - Brew install git - GUI Clients - ...


**Download for Windows**  
Click here to download the latest (2.41.0) 32-bit version of Git for ...

**Download for macOS**  
There are several options for installing Git on macOS. Note ...

**Documentation**  
Reference - What is Version Control? - Videos - External Links

[Más resultados de git-scm.com »](#)

Más preguntas



**Git**  
Software

Git es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia, la confiabilidad y compatibilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente. [Wikipedia](#)

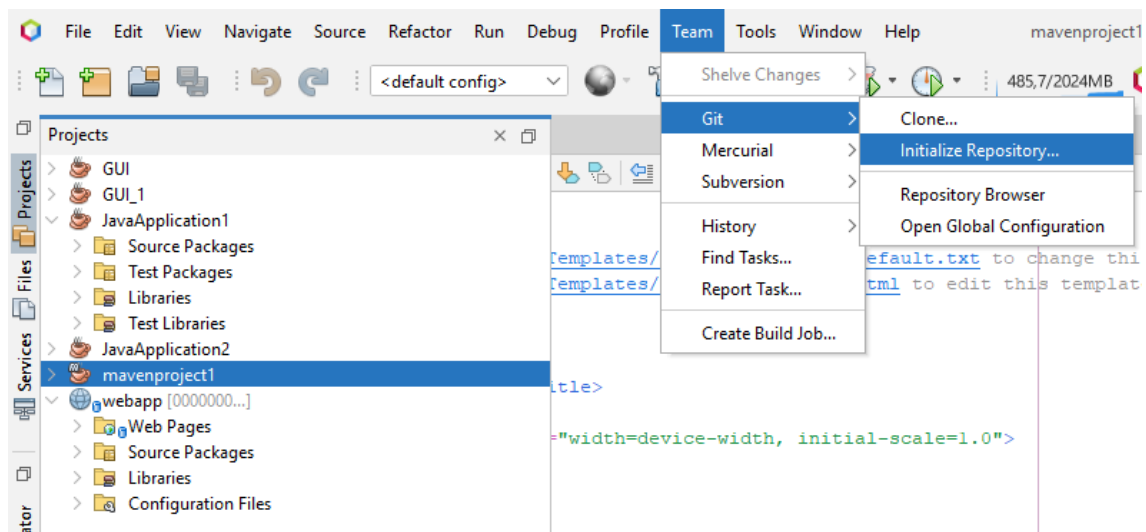
**Lenguajes de programación:** Python, C, Perl, C++, Script de shell, Tcl

**Fecha del lanzamiento inicial:** 7 de abril de 2005

**Desarrollador:** Junio Hamano, Linus Torvalds

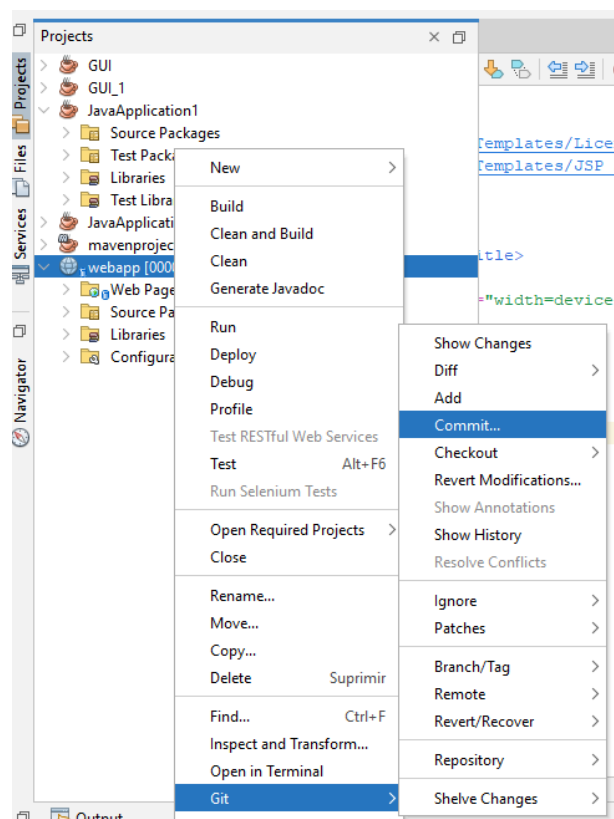


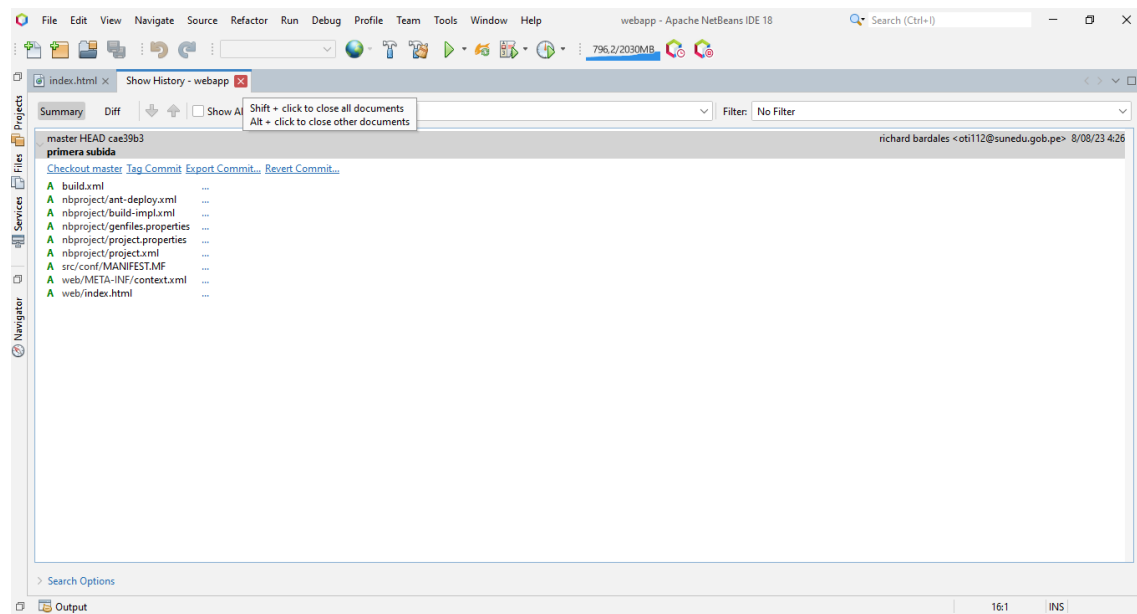
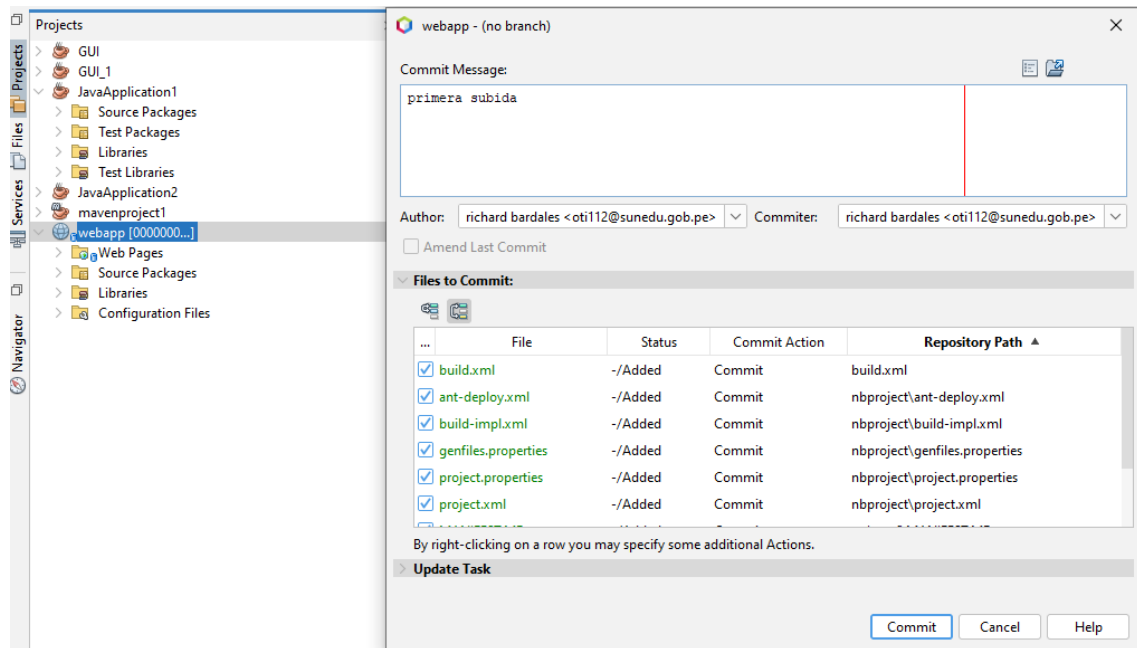
Inicializamos el repositorio.

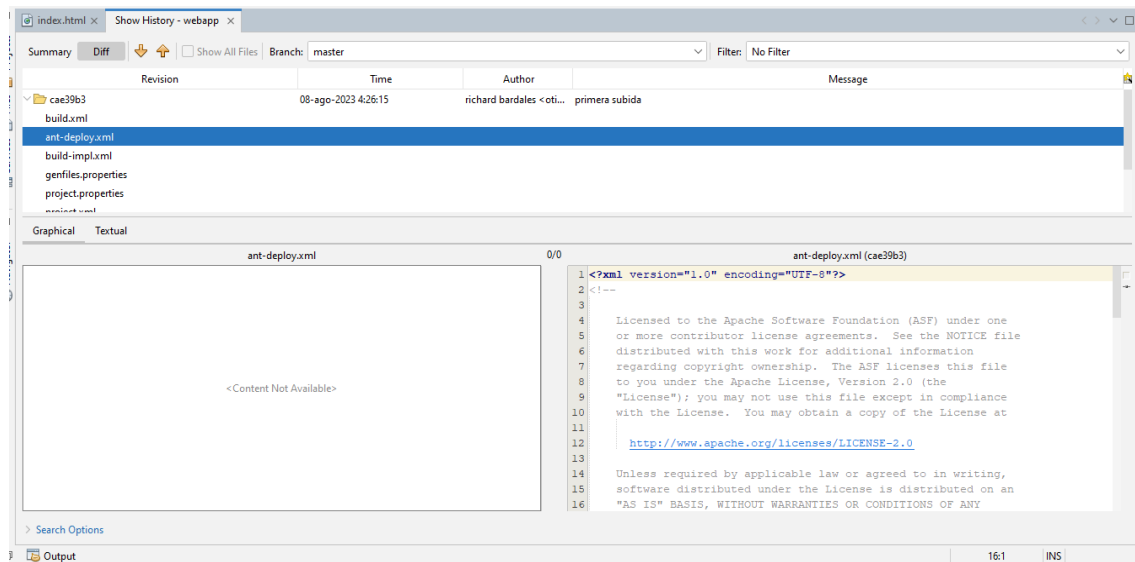


## Subir los cambios a través de GIT

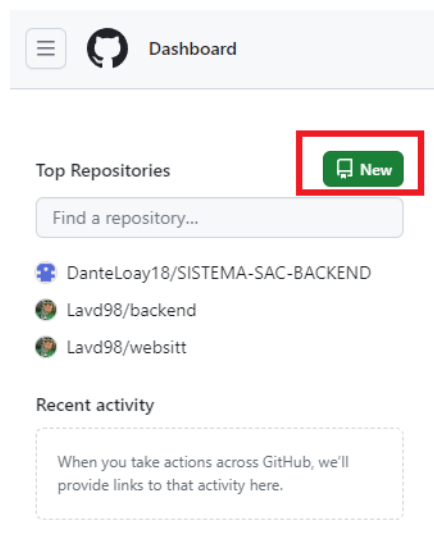
Agrupamos los cambios a través de un commit.









Creamos un nuevo repositorio



## Crear un nuevo repositorio


Un repositorio contiene todos los archivos del proyecto, incluido el historial de revisiones. ¿Ya tienes un repositorio de proyectos en otro lugar? [Importar un repositorio](#).

Los campos obligatorios están marcados con un asterisco (\*).

Dueño \* nombre del repositorio \*  
 punchito /   
 Java está disponible.

Los grandes nombres de repositorios son cortos y fáciles de recordar. ¿Necesitas inspiración? Qué tal si [amistoso-fagot](#) ?

Descripción (opcional)

- ☒  **Público**  
Cualquier persona en Internet puede ver este repositorio. Tú eliges quién puede comprometerse.
- ☐  **Privado**  
Tú eliges quién puede ver y comprometerse con este repositorio.

Inicialice este repositorio con:

- ☐ **Agregar un archivo README**  
Aquí es donde puede escribir una descripción larga para su proyecto. [Obtenga más información sobre los archivos README](#).

Agregar .gitignore


Plantilla .gitignore: Ninguno

Elija qué archivos no rastrear de una lista de plantillas. [Obtén más información sobre cómo ignorar archivos](#).

Elija una licencia



Licencia: Ninguno

Una licencia les dice a otros lo que pueden y no pueden hacer con su código. [Obtenga más información sobre las licencias](#).

 Estás creando un repositorio público en tu cuenta personal.

**Crear repositorio**

### Configuración rápida : si ha hecho este tipo de cosas antes

 Configurar en el escritorio o ☐ HTTPS ☒ SSH  

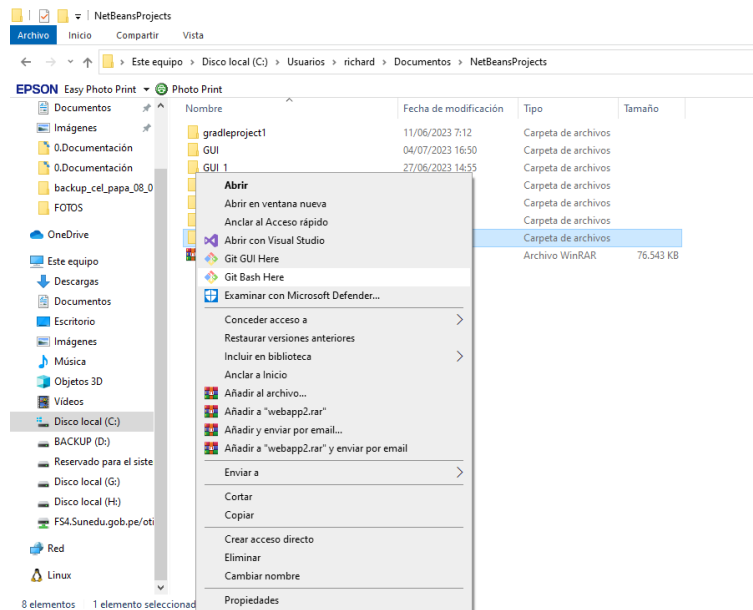
Comience creando un nuevo archivo o cargando un archivo existente. Recomendamos que cada repositorio incluya un README, LICENSE y .gitignore.

### ...o crea un nuevo repositorio en la línea de comando

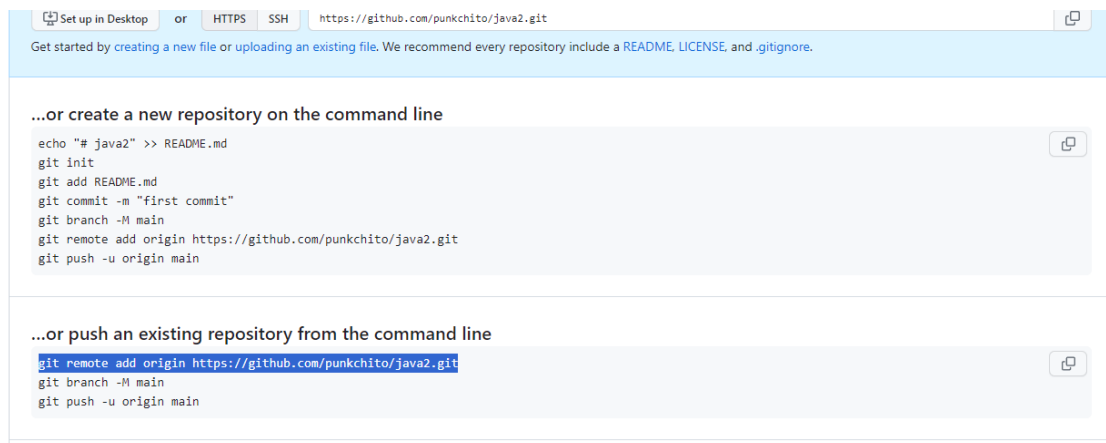
```
echo "# java" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/punchito/java.git
git push -u origin principal
```

Hacemos un Push a nuestro GITHUB

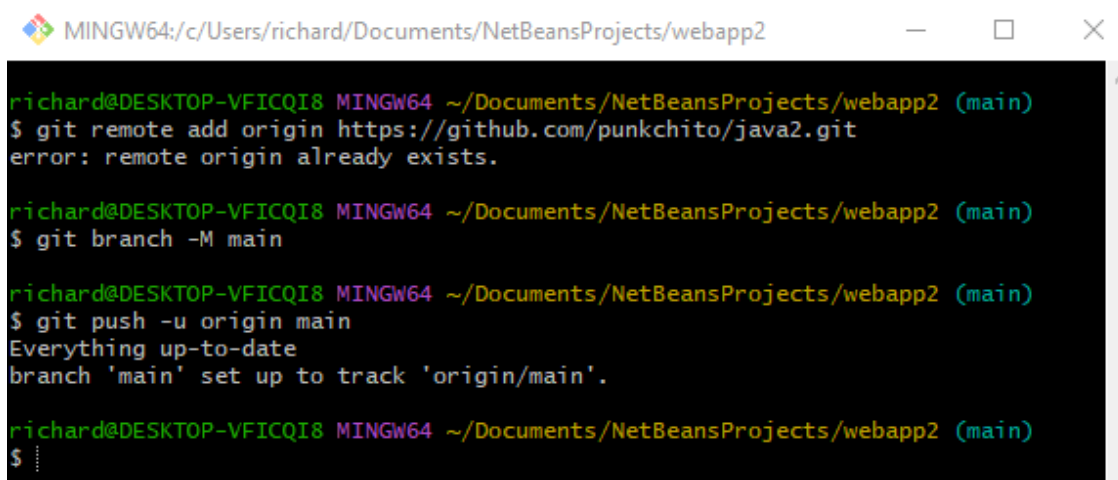
Primero abrimos nuestra carpeta donde se encuentra el proyecto con el GITBASH



Verificamos los comandos para ejecutar en el git bash, los cuales se encuentran en la sección code de nuestro repositorio



Procedemos a ejecutarlo en ese orden





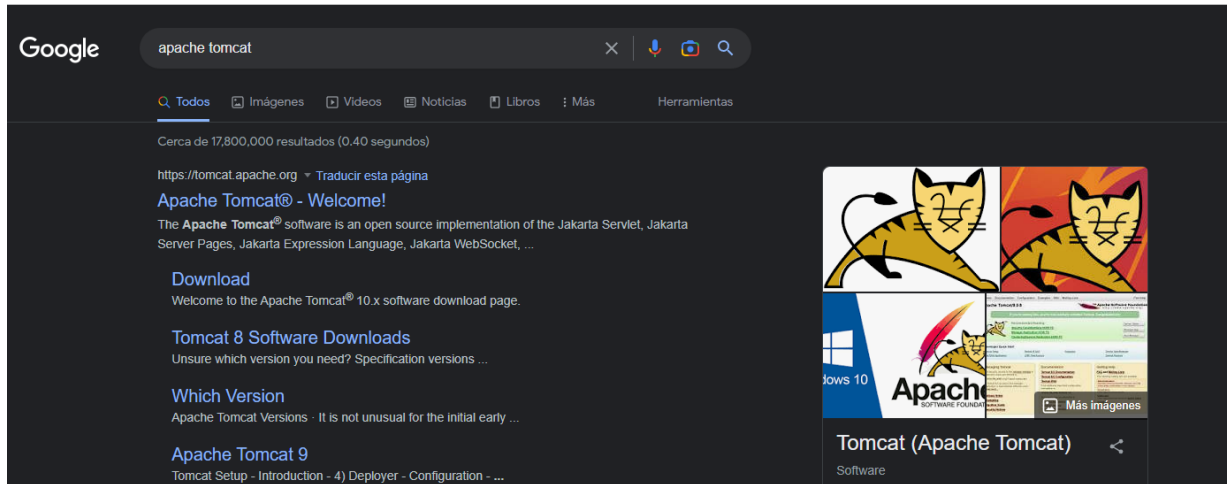
Nota: Si tenemos otro origen precargado podemos asignarle el nuevo origen de esta forma

```
git remote set-url origin git://new.url.here
```

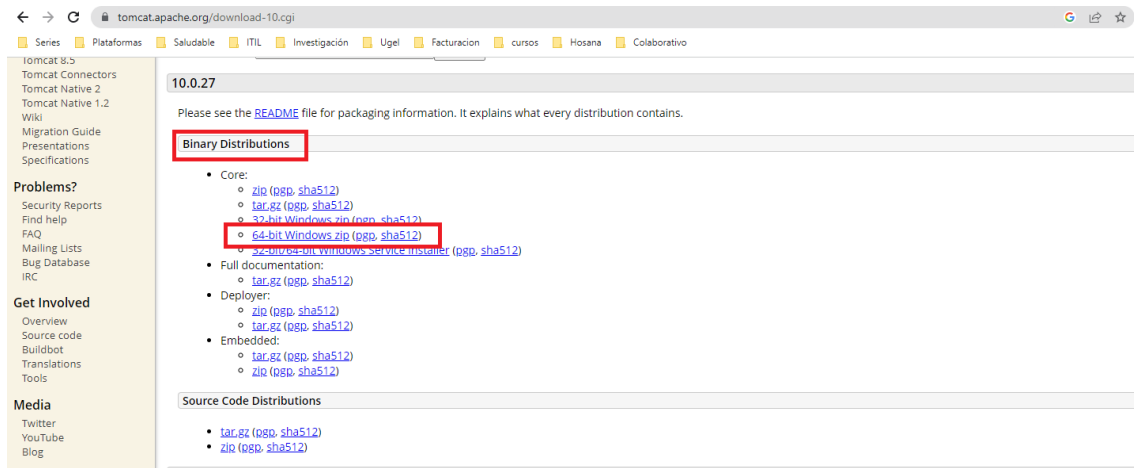
posteriormente hacer todo nuevamente

# Descargamos e instalamos el Apache Tomcat

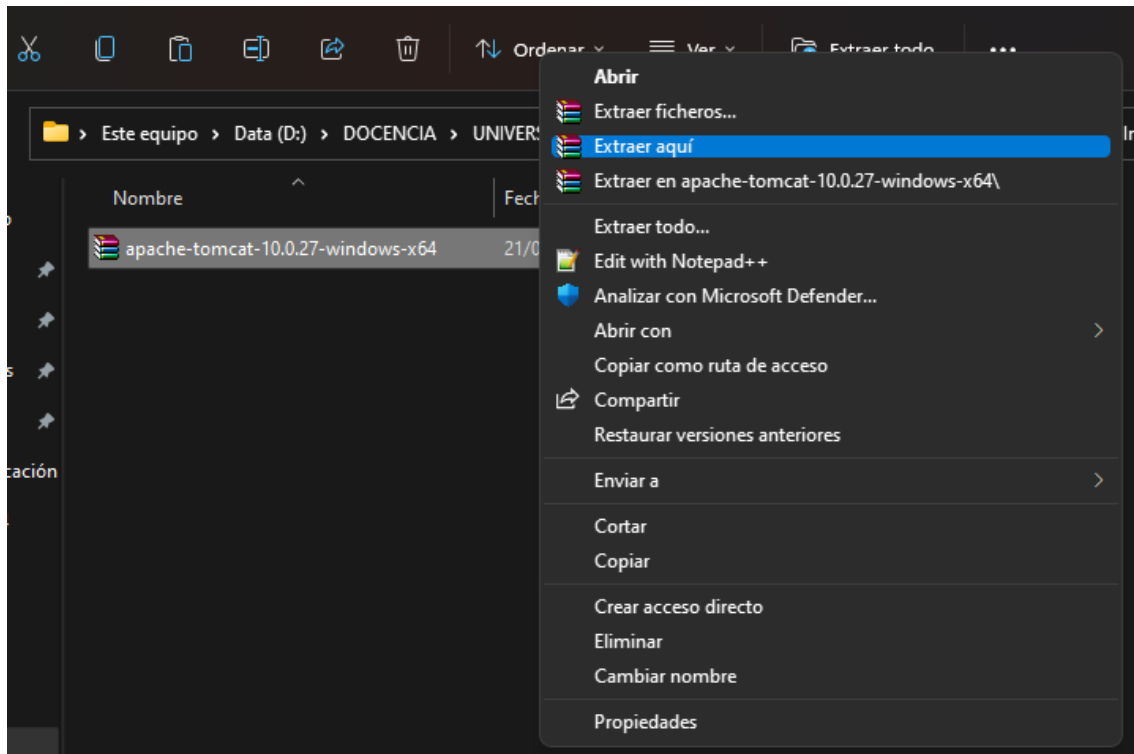
Nos dirigimos a google y buscamos tomcat y nos dirigimos al menú de descargas.



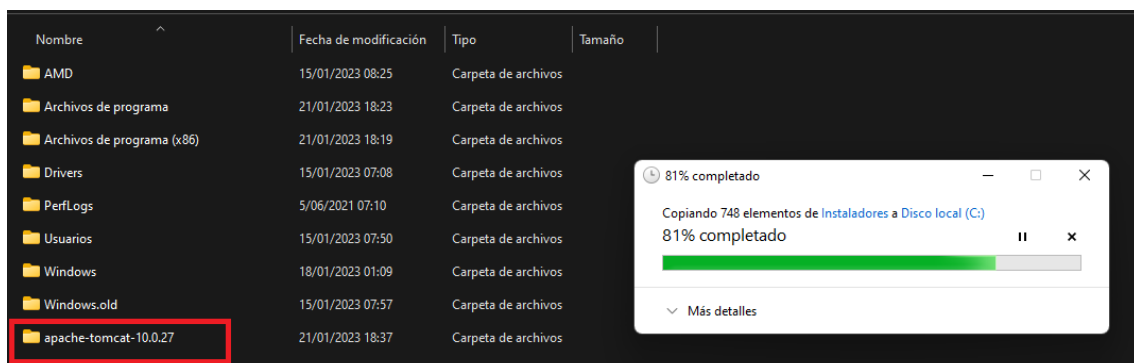
Nos dirigimos al apartado de Distribuciones Binarias y descargamos la versión para windows, puede ser la de 32 bits o la de 64 bits, dependiendo de la arquitectura del sistema operativo.



Extraemos el archivo

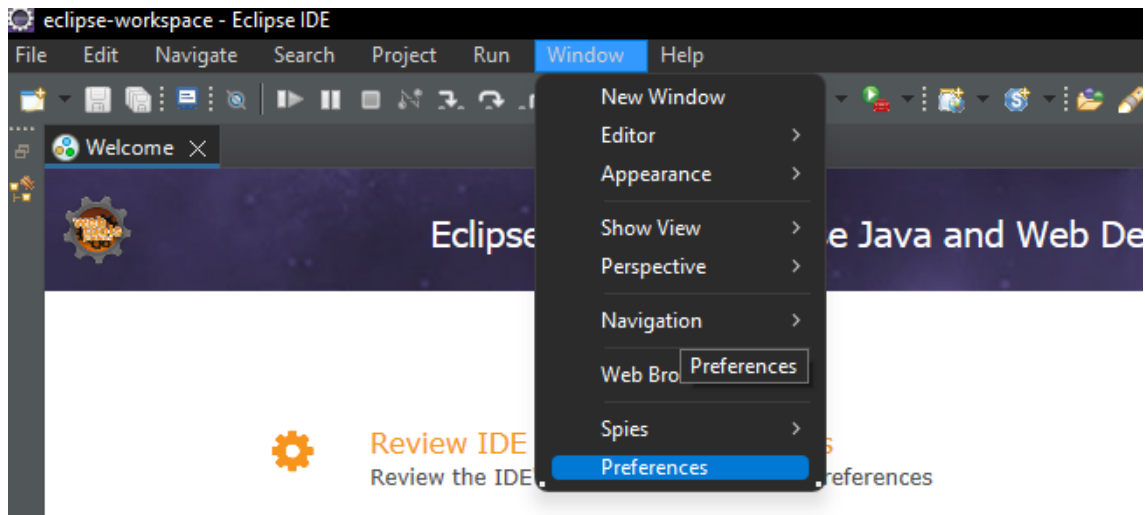


La carpeta extraída lo copiamos en el Disco C:/

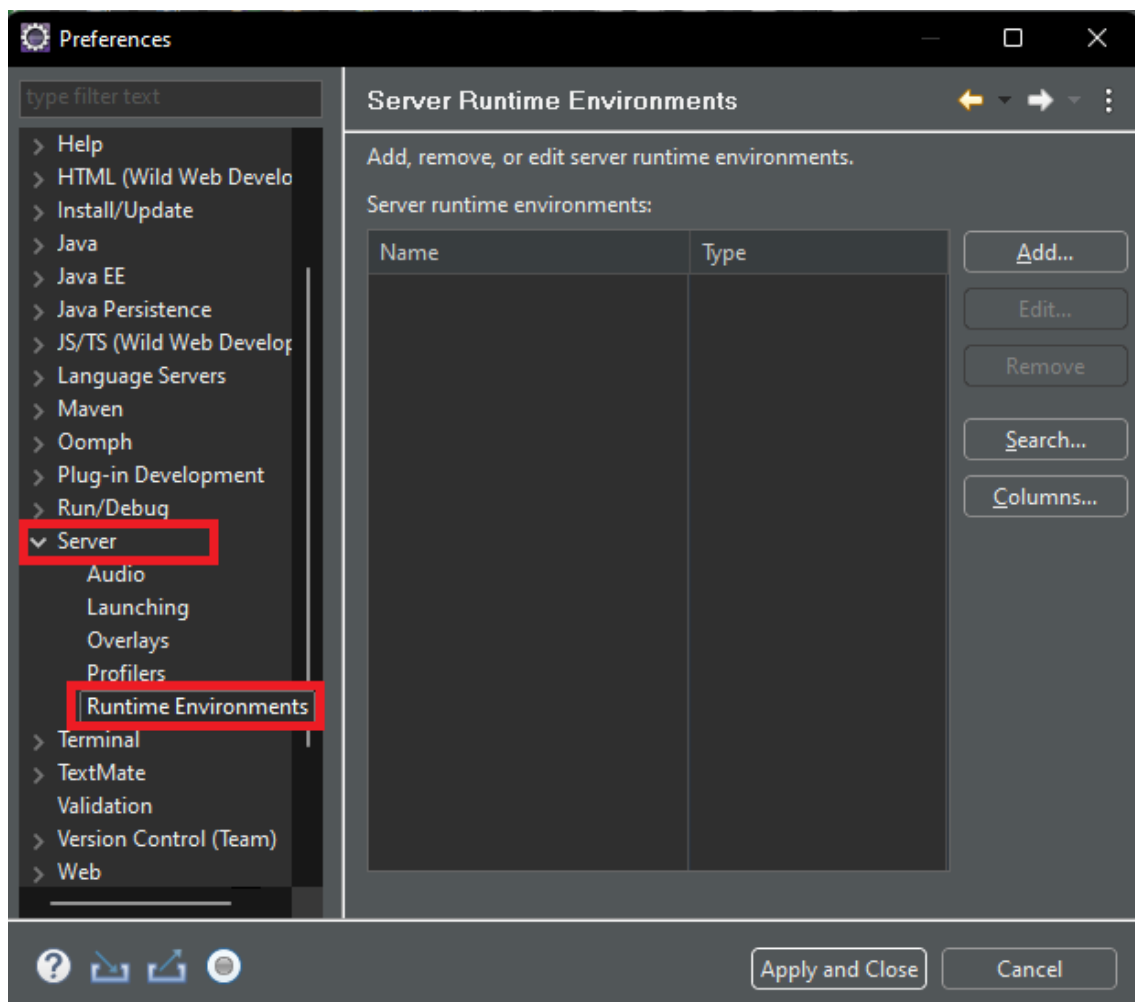


## Configuramos el tomcat para que ejecute aplicaciones web en Eclipse

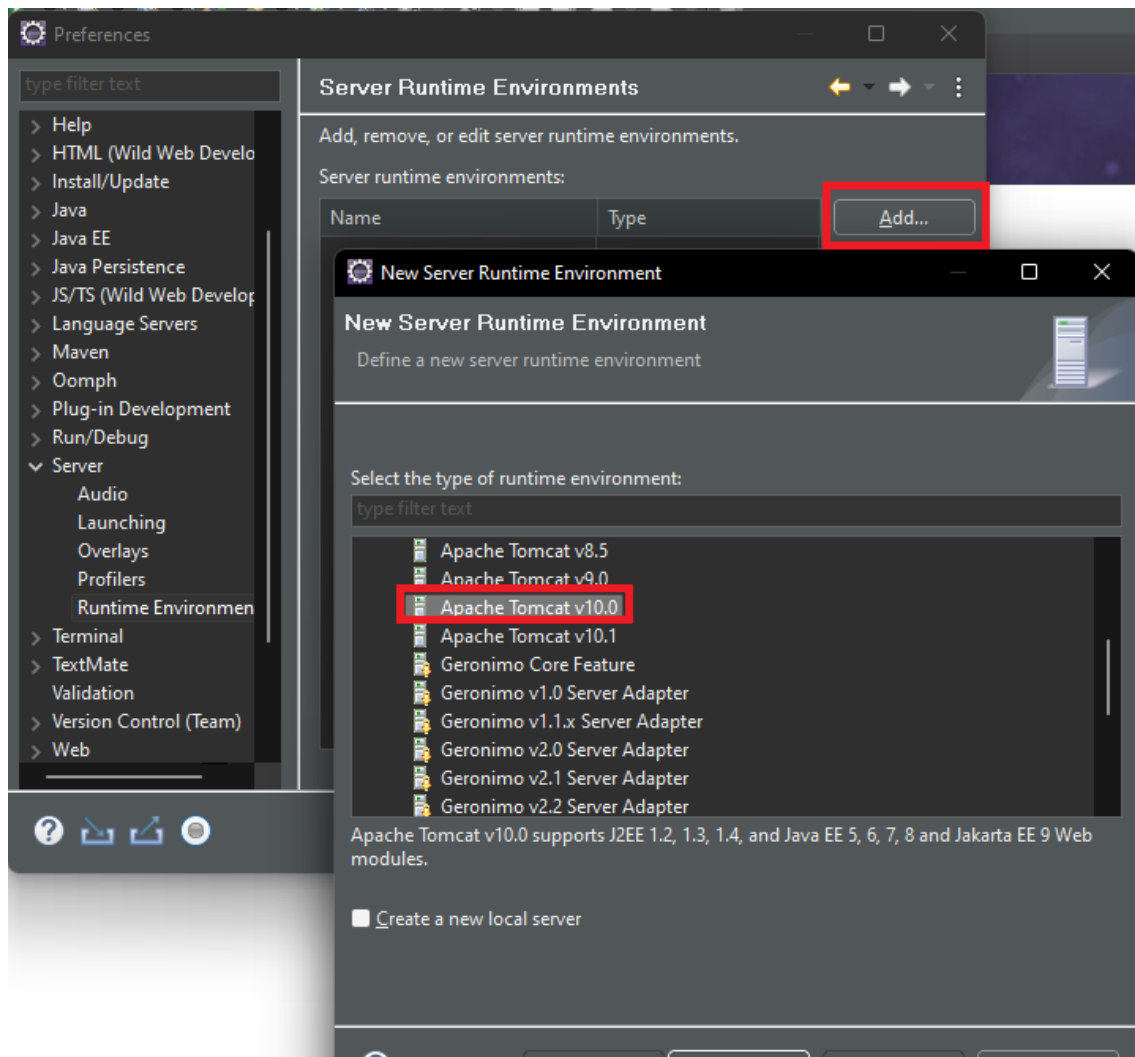
Nos dirigimos a eclipse a la opción **Windows->Preferences**.



Nos dirigimos a la sección **Server -> Runtime Enviroments**

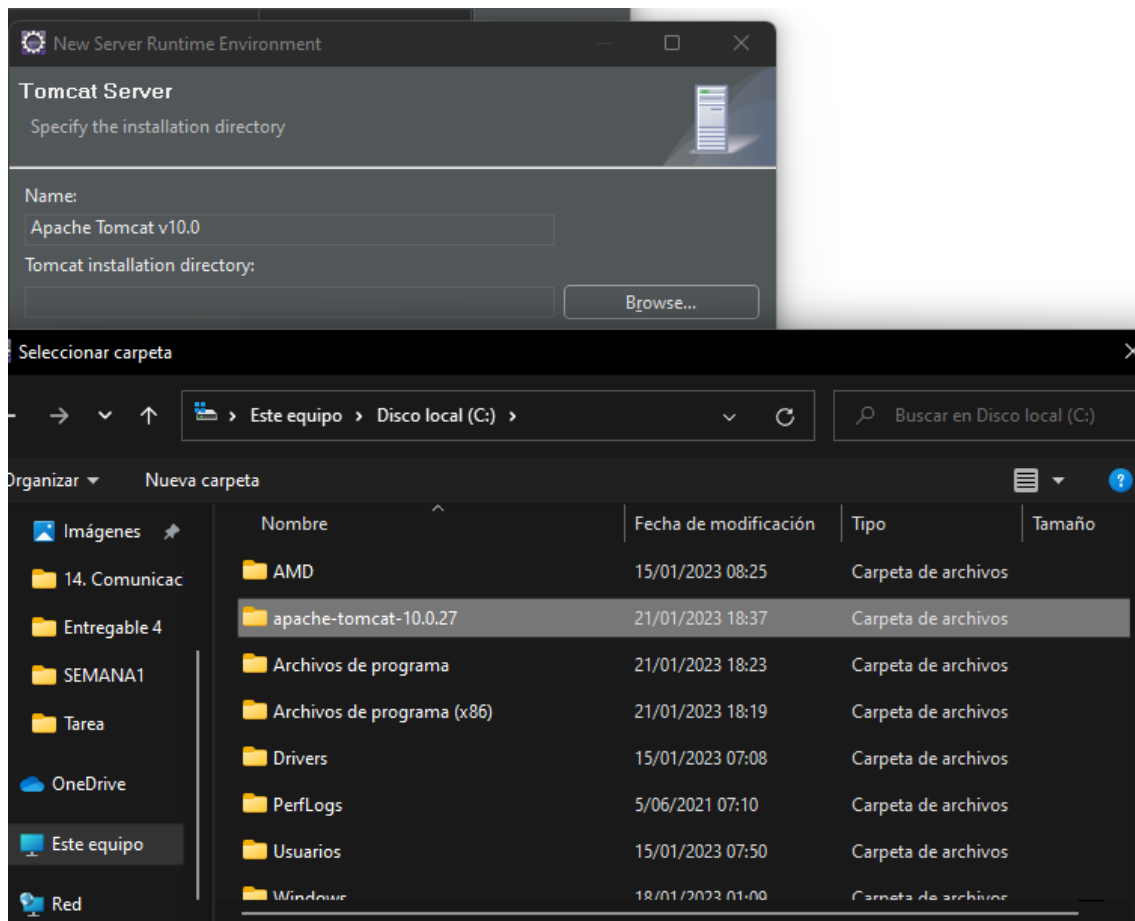


Damos clic en el botón **Add -> Apache Tomcat v10.0**

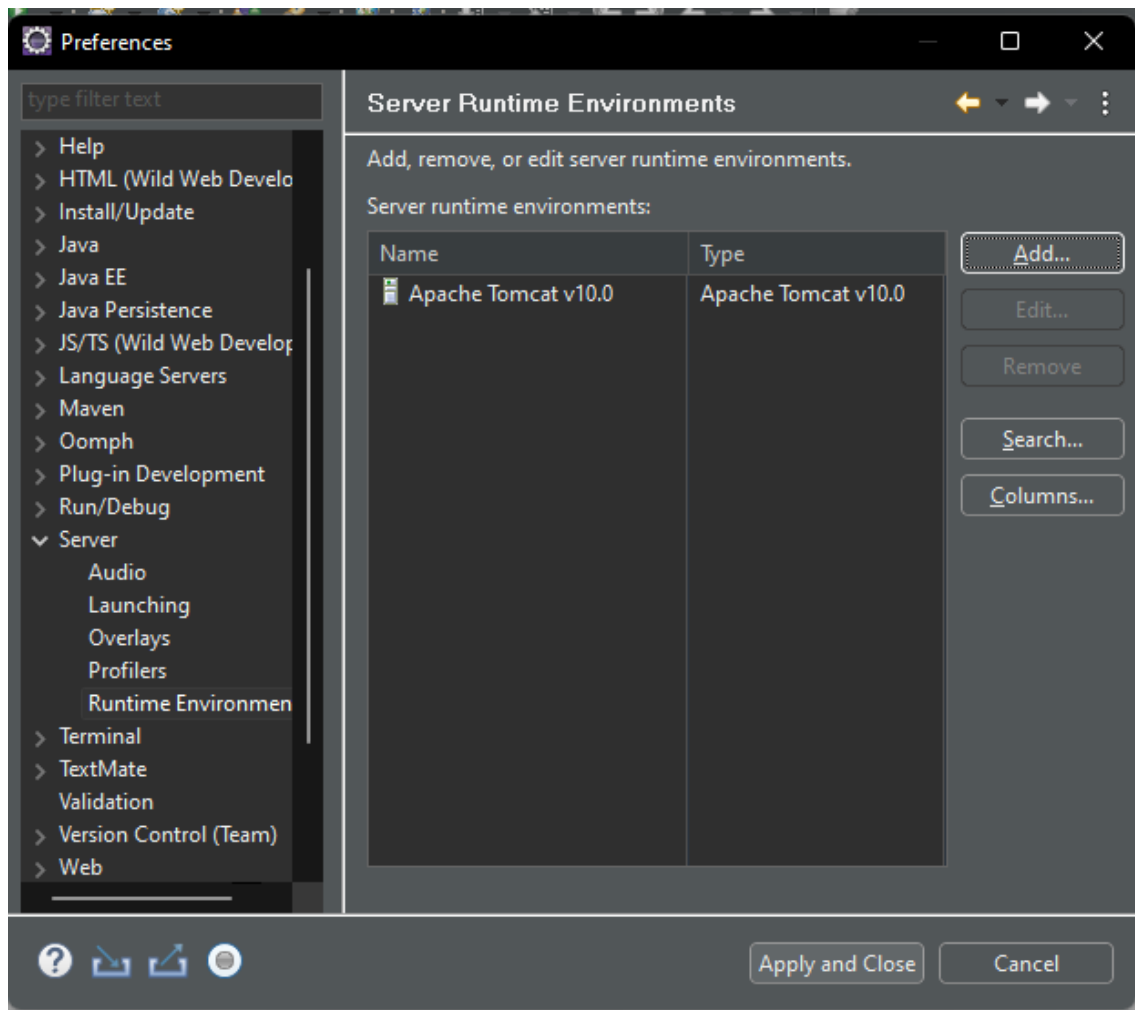


Damos clic en el botón **Browse** -> **Seleccionamos la ruta del tomcat** -> **Finish**

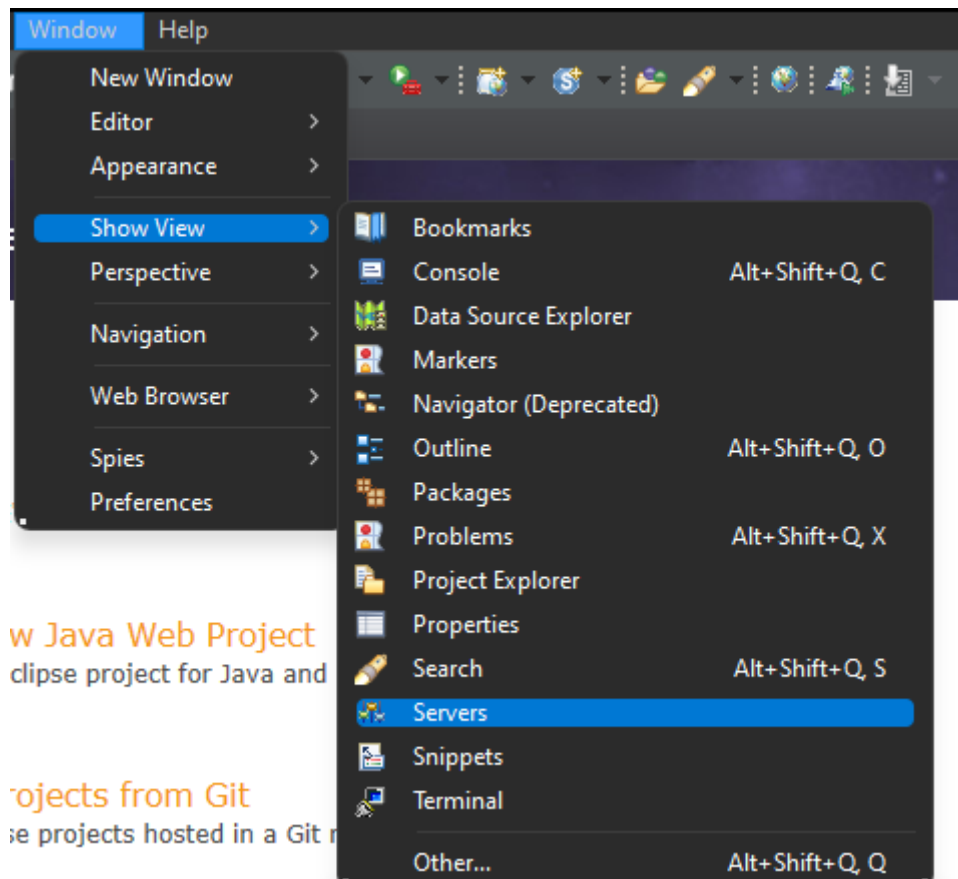




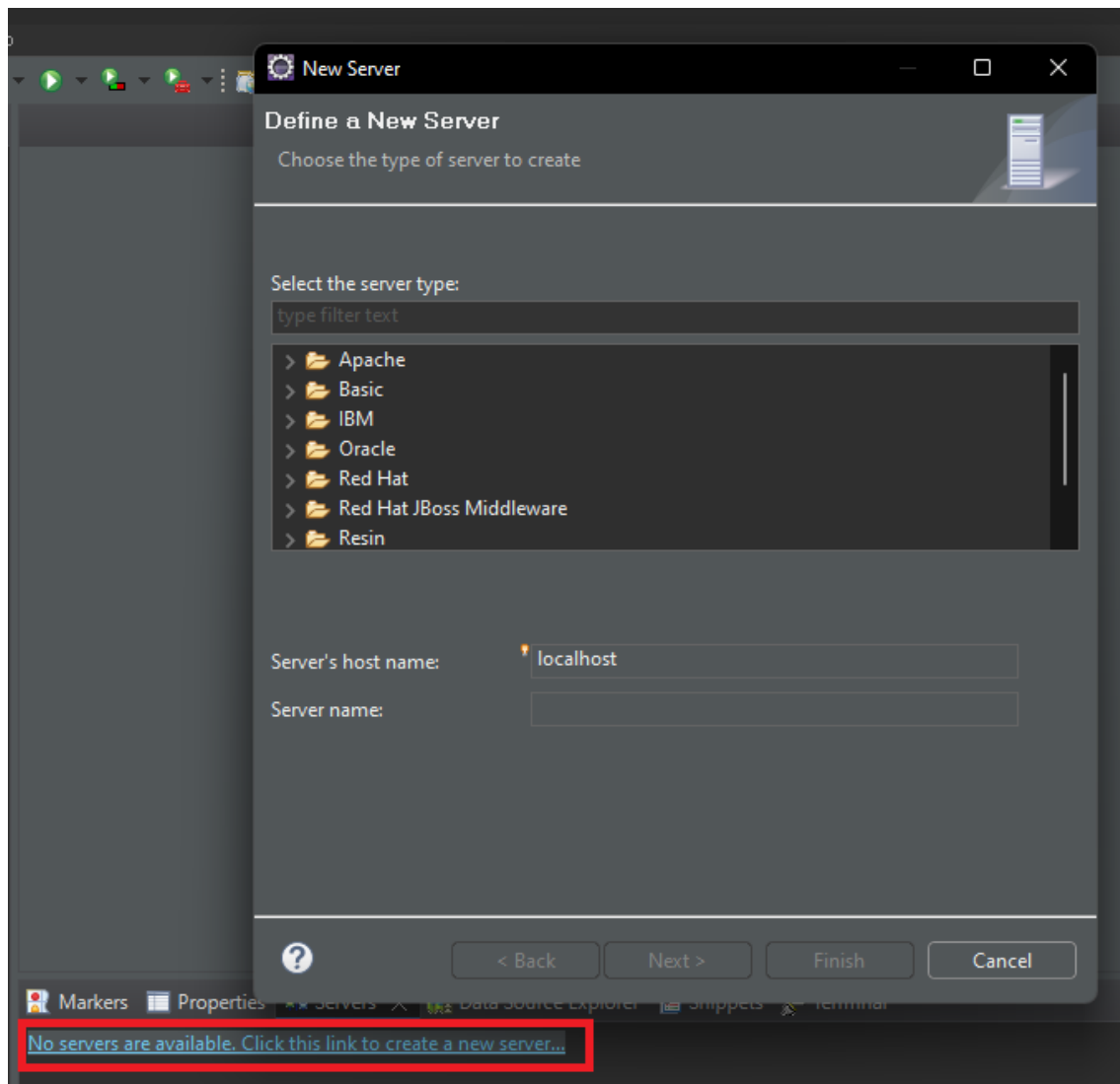
Clic en Apply and Closes



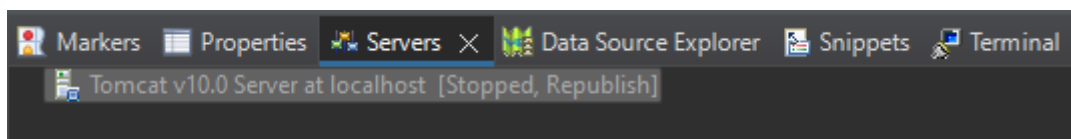
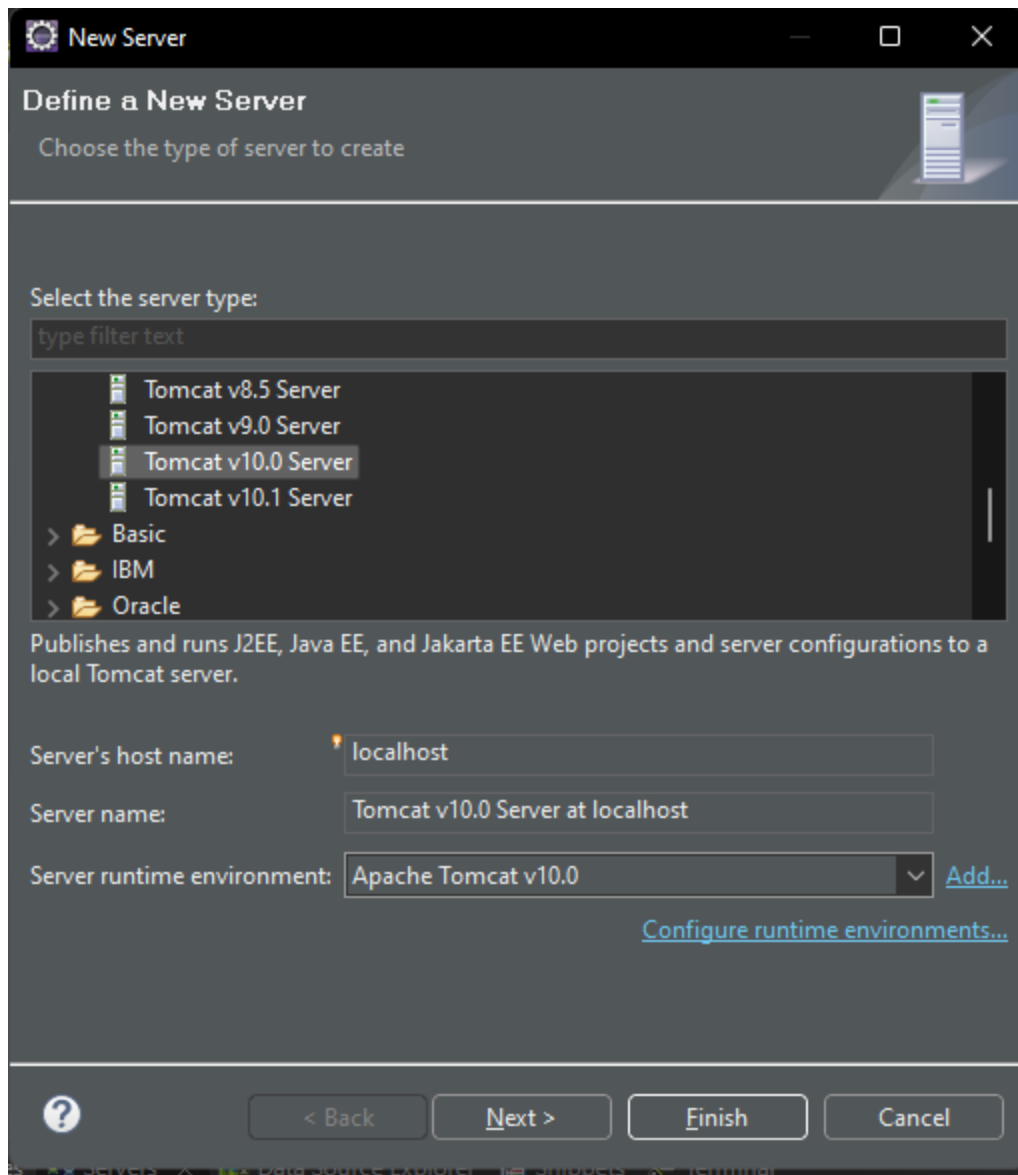
Nos dirigimos a windows - > show views -> servers



Le damos clic en nuevo servidor

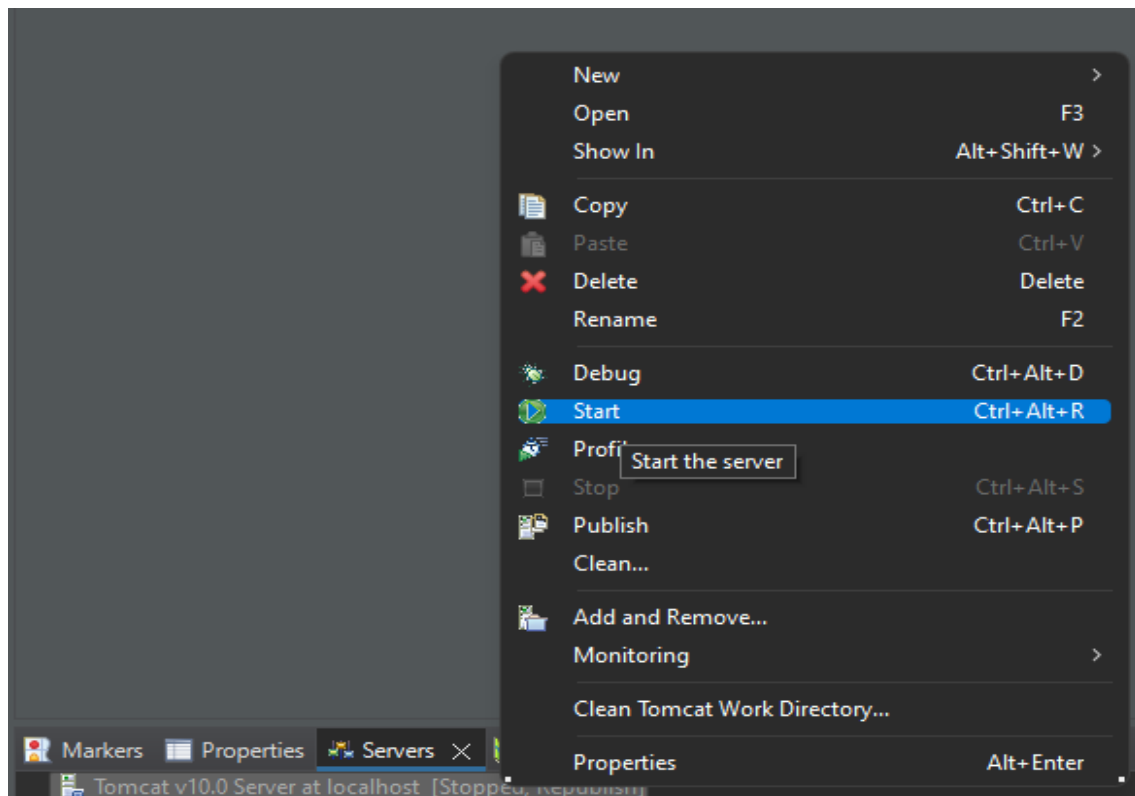


Seleccionamos tomcat y la versión 10.0

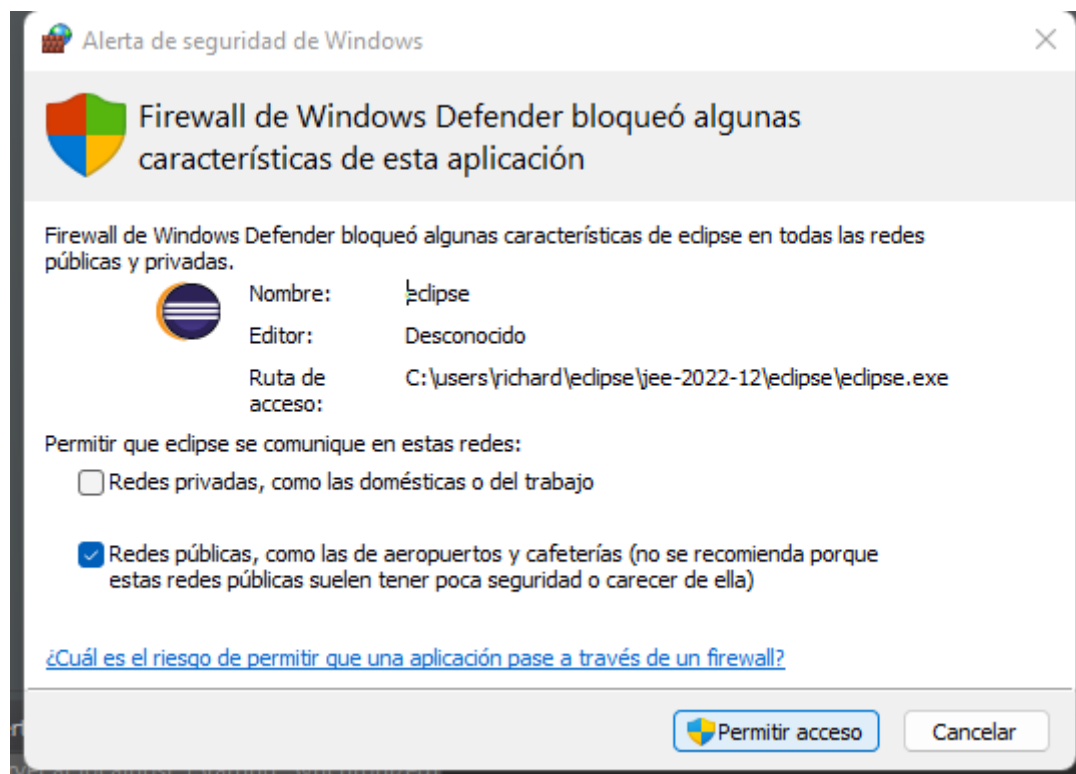


Seleccionamos el servidor, le damos clic derecho y luego le damos clic en start

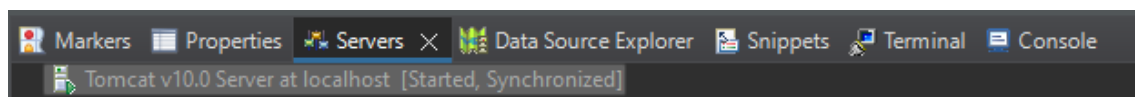




Permitimos el acceso



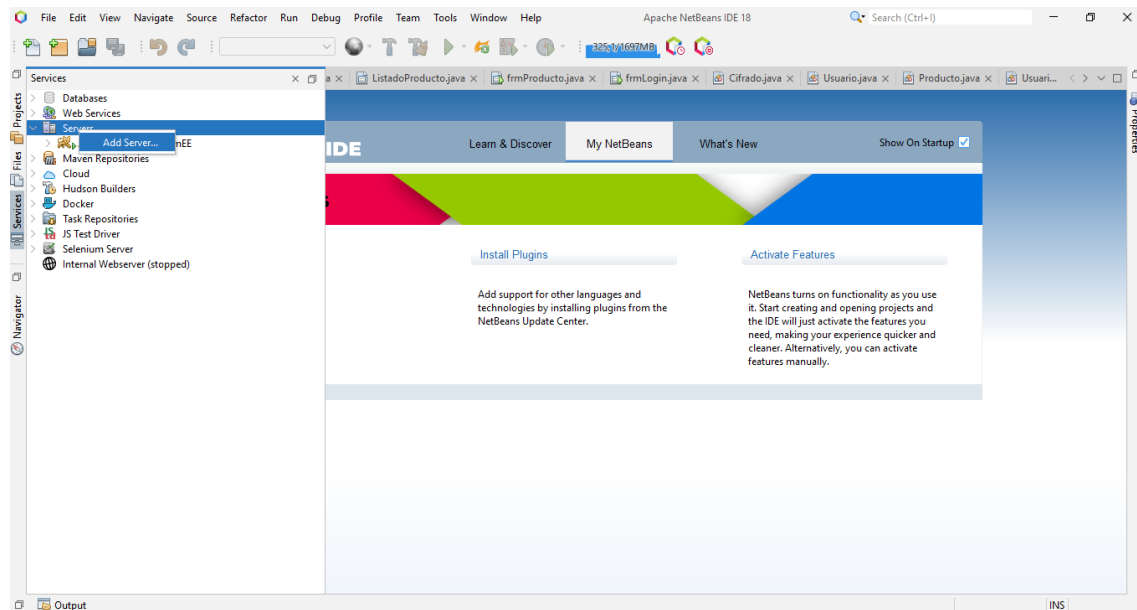
Nuestro servidor queda iniciado



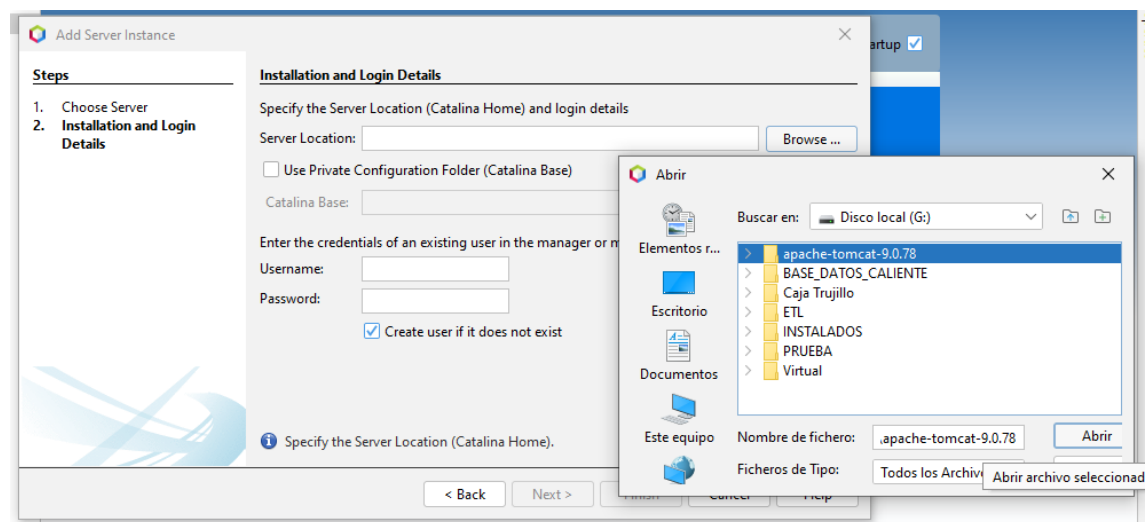
# Configuramos el tomcat para que ejecute aplicaciones web en NetBeans

Nos dirigimos a netbeans a la opción **Services->Server**

**Click en AddServer.**

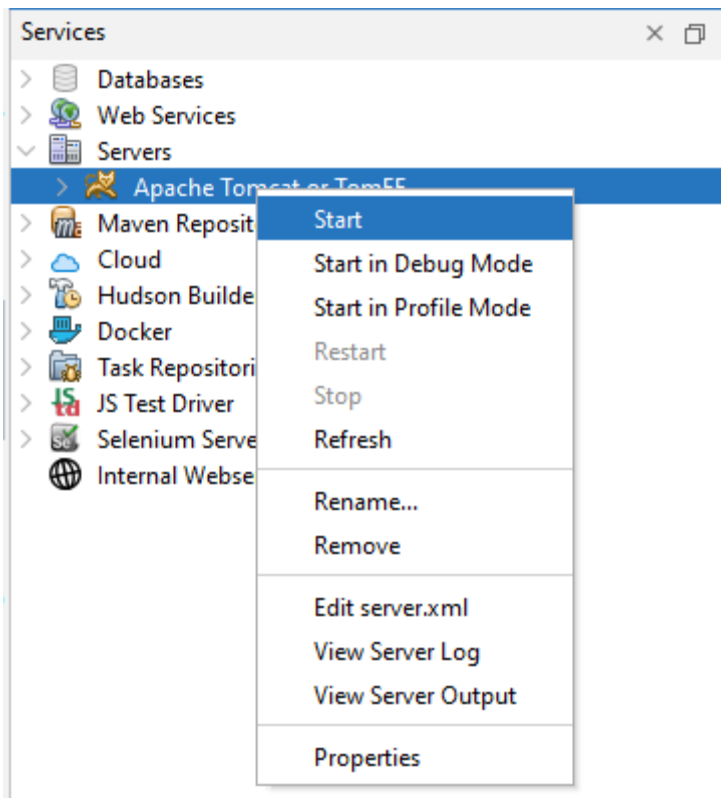


Luego seleccionar **Apache Tomcat or TomEE**, luego en Server **Location**, le damos clic en el botón **Browse** y finalmente ubicamos nuestra carpeta de tomcat descomprimido y luego clic en **Abrir**.

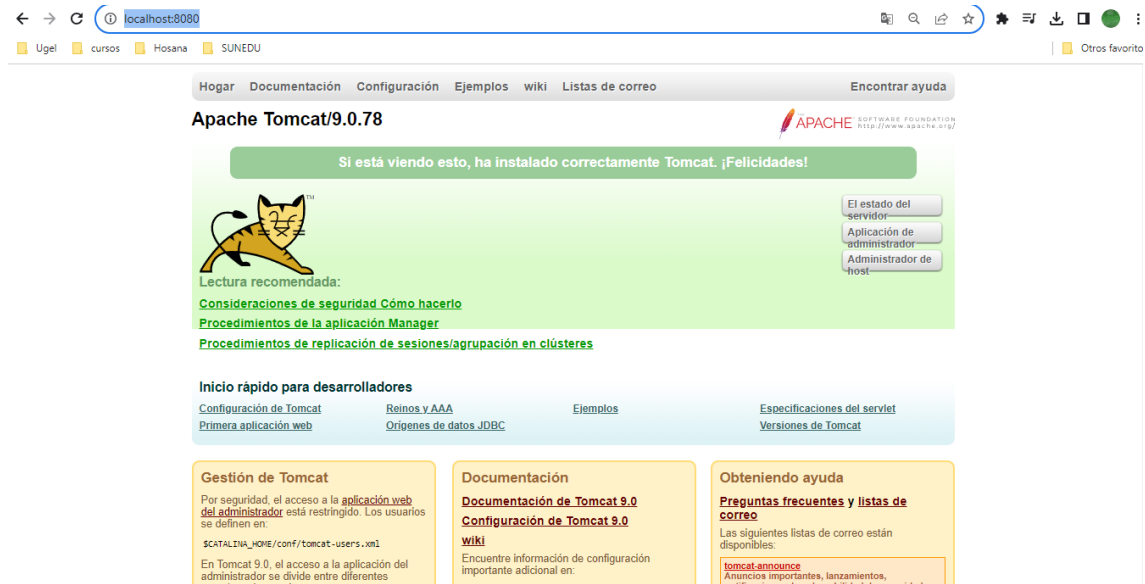


Luego colocamos un usuario y una contraseña para iniciar el servicio

Una vez configurado le damos clic en Start

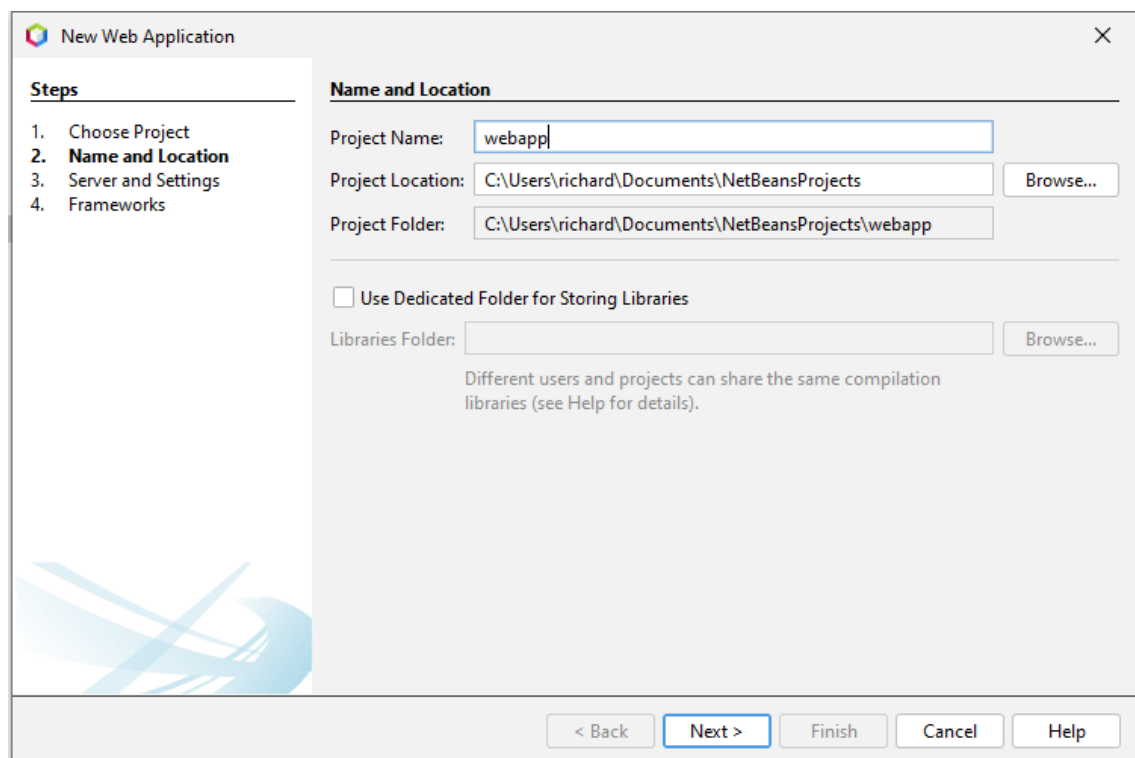
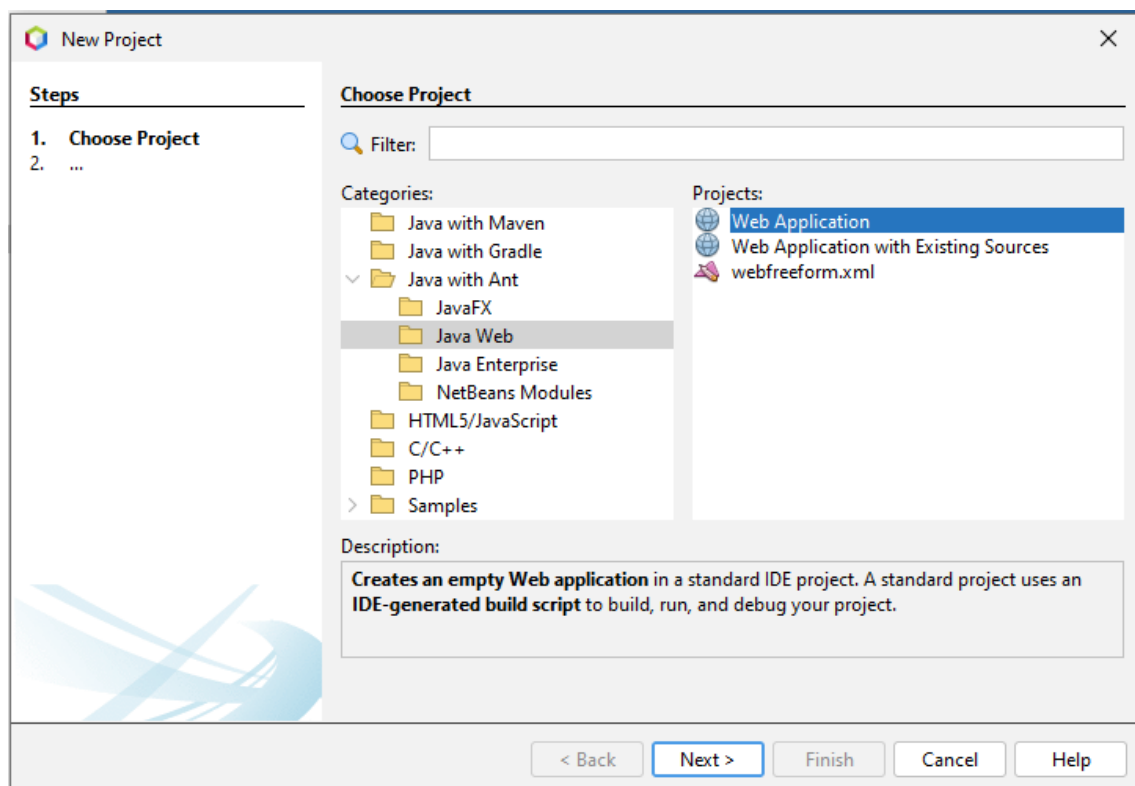
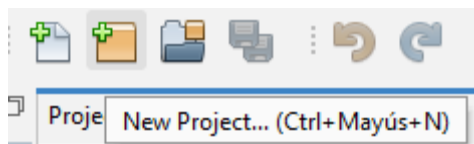


Finalmente probamos que el servicio se encuentre activo



## Iniciamos un proyecto web

Nos dirigimos a eclipse a la opción **New Project**.



New Web Application

Steps

1. Choose Project

2. Name and Location

3. **Server and Settings**

4. Frameworks

Server and Settings

Add to Enterprise Application: <None>

Server: Apache Tomcat or TomEE Add...

Java EE Version: Jakarta EE 8 Web

Context Path: /webapp

< Back

Next >

Finish

Cancel

Help

New Web Application

Steps

1. Choose Project

2. Name and Location

3. Server and Settings

4. **Frameworks**

Frameworks

Select the frameworks you want to use in your web application.

☐ Spring Web MVC

☐ JavaServer Faces

☐ Struts 1.3.10

< Back

Next >

Finish

Cancel

Help

index.html xRun Project (webapp) (F6)

SourceHistory

```
1 <!DOCTYPE html>
2 <!--
3 Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
4 Click nbfs://nbhost/SystemFileSystem/Templates/JSP\_Servlet/Html.html to edit this template
5 -->
6 <html>
7   <head>
8     <title>TODO supply a title</title>
9     <meta charset="UTF-8">
10    <meta name="viewport" content="width=device-width, initial-scale=1.0">
11  </head>
12  <body>
13    <div>TODO write content</div>
14  </body>
15 </html>
16
```