

# INDIVIDUAL REPORTS - E3.03

**Nombre:** José Cáceres Gómez

**DNI:** 44248655J

**Correo:** [joscacgom@alum.us.es](mailto:joscacgom@alum.us.es)

## Índice

|                                  |          |
|----------------------------------|----------|
| <b>Development configuration</b> | <b>2</b> |
| Introducción                     | 2        |
| Java                             | 2        |
| Firefox & Gecko driver           | 2        |
| Maven                            | 3        |
| Maria DB & DBeaver               | 3        |
| Eclipse & Lombok                 | 4        |
| Anexo                            | 4        |
| <b>Describing a WIS</b>          | <b>5</b> |
| <b>Testing a WIS</b>             | <b>5</b> |


# Development configuration

## Introducción

Para el desarrollo de la asignatura “Diseño y Pruebas 2” es necesario tener un entorno de trabajo determinado para que funcione correctamente el proyecto base con el que trabajaremos a lo largo de la asignatura.

## Java

Lo primero que necesitábamos instalar es “Java jdk 8 Update 281” que será el lenguaje de programación en el que está el proyecto base. Para la comprobación de la correcta instalación es necesario ejecutar los comandos “java -version” y “javac -version”

 Símbolo del sistema


```
Microsoft Windows [Versión 10.0.19042.1526]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\jose9>java -version
java version "1.8.0_281"
Java(TM) SE Runtime Environment (build 1.8.0_281-b09)
Java HotSpot(TM) 64-Bit Server VM (build 25.281-b09, mixed mode)

C:\Users\jose9>javac -version
javac 1.8.0_281
```

## Firefox & Gecko driver

Para realizar las pruebas necesitamos instalar el navegador Firefox y , gracias a Gecko Driver podremos automatizarlas. Para comprobar su correcta instalación debemos escribir los comandos “firefox --version | more” y “geckodriver --version”

 Símbolo del sistema

```
C:\Users\jose9>firefox --version | more
Mozilla Firefox 97.0.1

C:\Users\jose9>geckodriver --version
geckodriver 0.30.0 (d372710b98a6 2021-09-16 10:29 +0300)

The source code of this program is available from
testing/geckodriver in https://hg.mozilla.org/mozilla-central.

This program is subject to the terms of the Mozilla Public License 2.0.
You can obtain a copy of the license at https://mozilla.org/MPL/2.0/.
```

## Maven

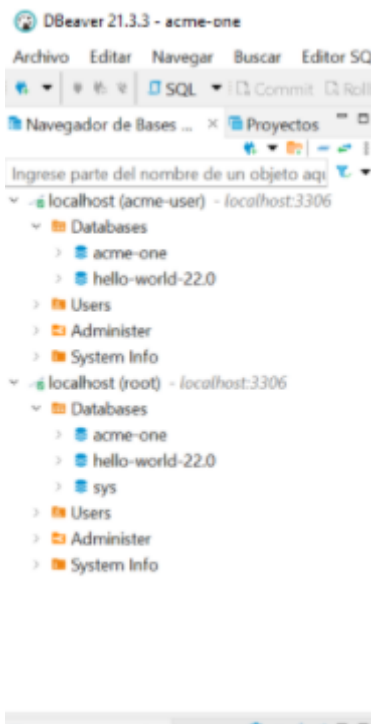
Para administrar nuestros componentes de software y empaquetar nuestros proyectos, trabajaremos con Maven. Es una herramienta de administración de compilaciones. Comprobamos su correcta instalación ejecutando el comando “mvn -version”.

 Símbolo del sistema

```
C:\Users\jose9>mvn -version
Apache Maven 3.8.4 (9b656c72d54e5bacbed989b64718c159fe39b537)
Maven home: C:\Users\jose9\OneDrive\Escritorio\Workspace-22.0\Tools\Infrastructure\maven-3.8.4
Java version: 1.8.0_281, vendor: Oracle Corporation, runtime: C:\Program Files\Java\jdk1.8.0_281\jre
Default locale: es_ES, platform encoding: Cp1252
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"
```

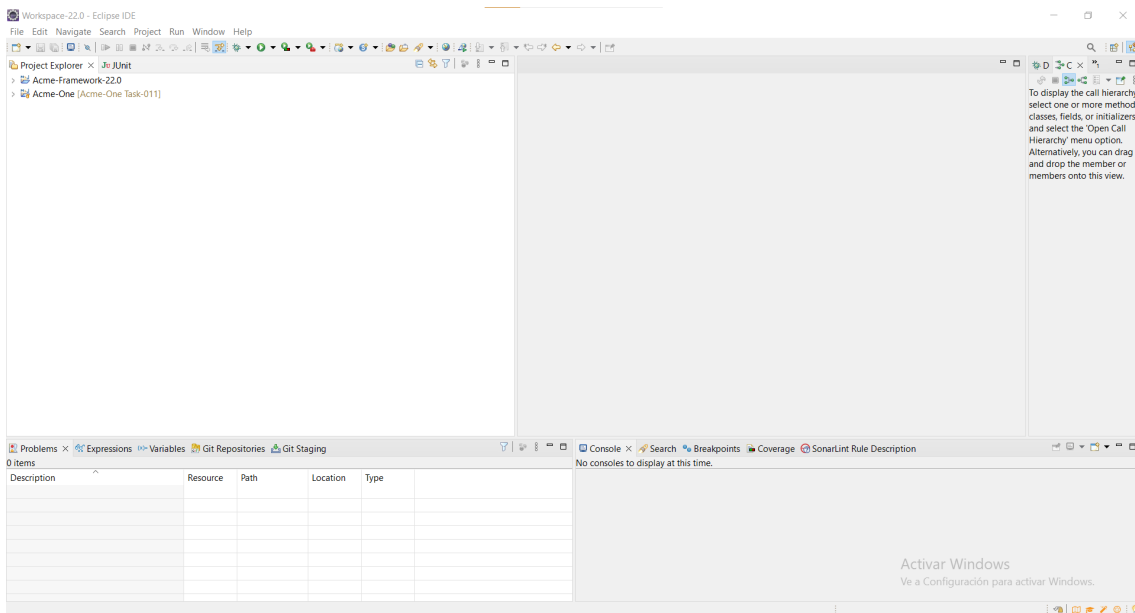
## Maria DB & DBeaver

Usaremos MariaDB, que es el servidor de la base de datos. Como herramienta de gestión de base de datos usaremos DBeaver. Para que ambas aplicaciones funcionen es necesario primero ejecutar MariaDB y posteriormente DBeaver. Para comprobar que la conexión se ha realizado correctamente en DBeaver se debe mostrar un tick verde sobre la conexión creada (en el ejemplo localhost:3306).



## Eclipse & Lombok

Para escribir y depurar nuestros proyectos, la herramienta utilizada será Eclipse. Entorno de desarrollo integrado (IDE). Además, gracias a Lombok, conseguimos simplificar la escritura de código Java, y varios complementos que simplifican el desarrollo web. Para que Eclipse funcione correctamente es necesario usar el workspace proporcionado en la asignatura, el java jdk que se nos indica en las transparencias e importar el archivo de preferencias "Eclipse-Preferences.epf" que lo encontramos también en la carpeta que nos proporciona la asignatura.



## Anexo

Todas las herramientas que utilizaremos en esta asignatura las hemos instalado a partir de la carpeta "Tools" que viene en el fichero "Workspace-22.0" descargado desde la página web de la asignatura.

## Describing a WIS

Un sistema de información web está compuesto por componentes, que con un objetivo común, interactúan entre ellos, ayudando a administrar, almacenar y distribuir información relevante para los procesos fundamentales de una organización, en función de sus objetivos.

## Testing a WIS

El testing es la herramienta que nos permite detectar los errores en el proceso de desarrollo para poder solventarlos a tiempo. Es un proceso paralelo al de desarrollo, ya que a medida que avanzamos con el proyecto aparecerán errores que corregir a medida que implementamos más funcionalidad. Hoy en día se usan pruebas automatizadas ahorrando de esta manera tiempo del desarrollo del proyecto y mejorando la fiabilidad, integración continua y se persigue una gran cobertura de pruebas. Hay muchos tipos de pruebas, desde las más básicas como los test unitarios, de integración, aceptación hasta los End2End que prueban todo el sistema en su conjunto.