

INDIVIDUAL REPORTS - E3.03

Nombre: Jorge Toledo Vega

DNI: 49520347J

Correo: jortolhur@alum.us.es

Índice

Development configuration	2
Introducción	2
Java	2
Firefox & Gecko driver	2
Maven.....	3
Maria DB & DBeaver	3
Eclipse & Lombok	4
Anexo	4
Describing a WIS.....	5
Testing a WIS.....	5

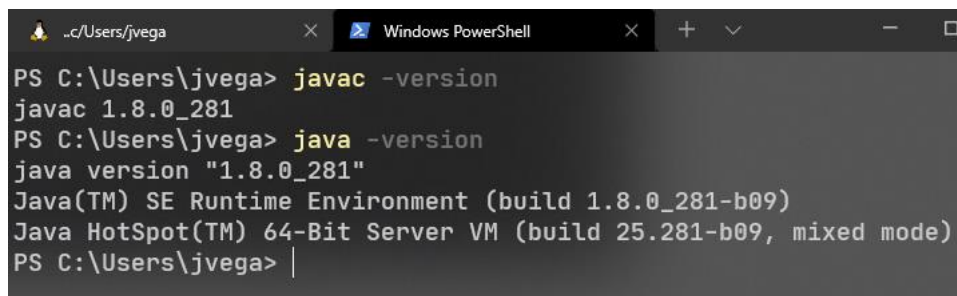
Development configuration

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

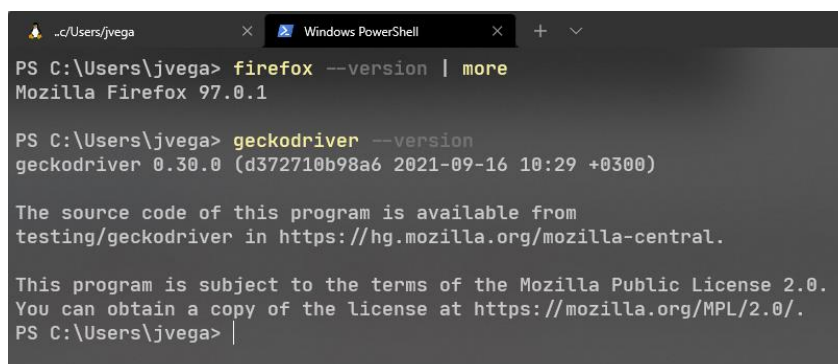
Lo primero que necesitábamos instalar es “Java jdk 8 Update 281” que será el lenguaje de programación en el que está el proyecto base. Para la comprobación de la correcta instalación es necesario ejecutar los comandos “java -version” y “javac -version”



```
PS C:\Users\jvega> javac -version
javac 1.8.0_281
PS C:\Users\jvega> 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)
PS C:\Users\jvega> |
```

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”



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

PS C:\Users\jvega> 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/.
PS C:\Users\jvega> |
```

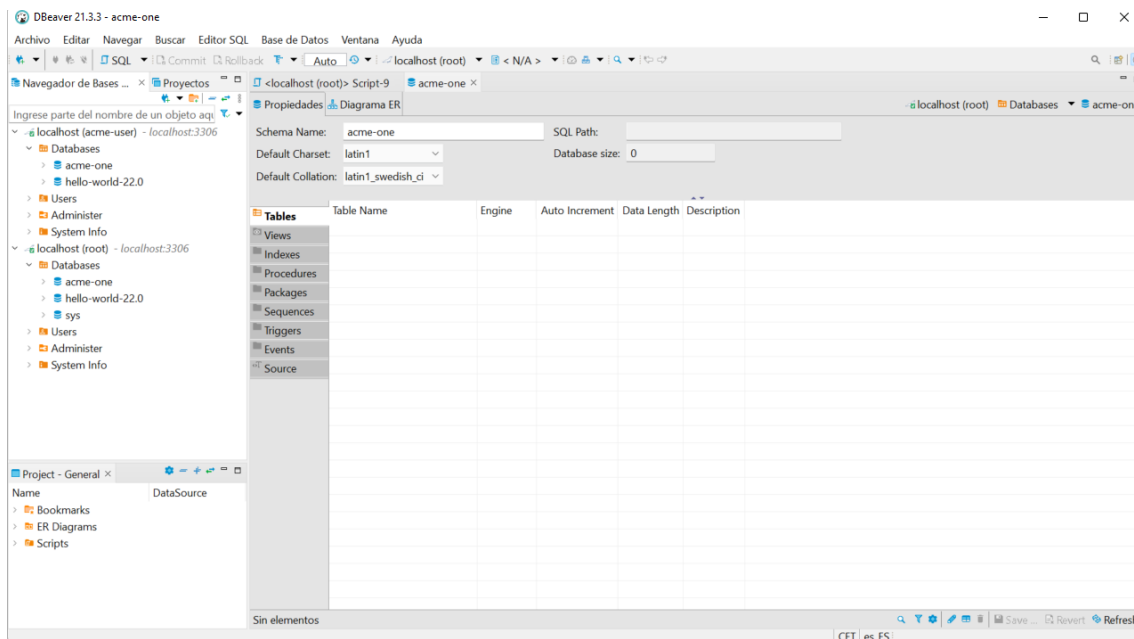
Maven

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”.

```
PS C:\Users\jvega> mvn -version
Apache Maven 3.8.4 (9b656c72d54e5baced989b64718c159fe39b537)
Maven home: C:\Users\jvega\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"
PS C:\Users\jvega> |
```

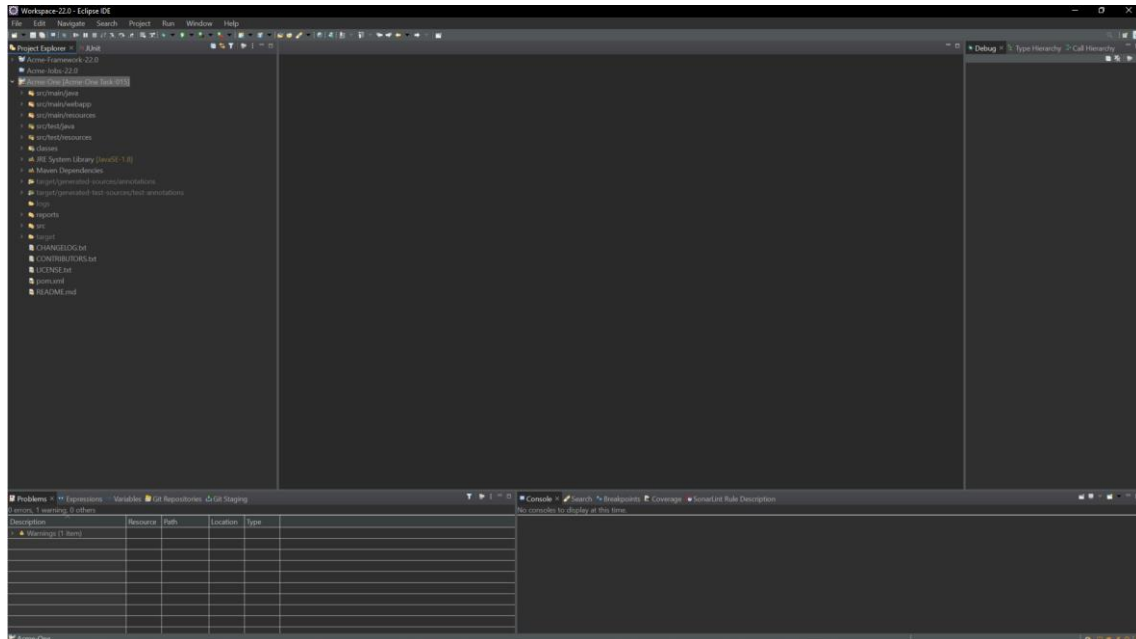
Maria DB & DBeaver

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

Los sistemas de información web o sistemas informáticos, se entienden como sistemas que están compuestos por diferentes componentes que, interactúan o no entre sí, formando un sistema capaz de llevar a cabo el objetivo de negocio de una organización o un particular.

Testing a WIS

La importancia de los test para un sistema de información web es muy importante, debido a que los test comprueban todas las funcionalidades del sistema, lo que nos permite conforme realizamos actualizaciones de nuestro sistema, determinar y prevenir que posibles fallos se introduzcan en producción. Por ello es muy interesante, y casi obligatorio hoy en día, hacer uso de mecanismos de automatización de tests y despliegue, y así evitar que se actualice nuestra versión en producción con errores.

Existen muchos tipos de test entre los más comunes son los test unitarios, que prueban de manera muy aislada una determinada funcionalidad, y los test E2E (end to end) que prueban el sistema web de manera más completa, y normalmente comprueban varias historias de usuario juntas, por ejemplo -> Usuario se registra, usuario sube foto a la web, usuario edita la información de la foto y usuario le comparte la foto a otro usuario.