

## 7.1. Instalación de *Tomcat 7.0* en *Windows*

En esta práctica se instalará el servidor de aplicaciones *Apache Tomcat*, en su versión 7.0, en la máquina **DesarrolloW7**.

*Apache Tomcat 7.0* requiere para su funcionamiento tener instalada la máquina virtual de Java (*Java Standard Edition Runtime Environment (JRE)*) en su versión 6.0 o posterior.

### 1. Instalación de *Java Standard Edition Development Kit (JDK)*.

- 1.1. Inicia sesión en el equipo **DesarrolloW7XX** con el usuario **alumno**.
- 1.2. Descarga *Java SE Development Kit* de <http://www.oracle.com/technetwork/java/javase/downloads/index.html>. En el momento de realizar este manual estaba disponible la versión 7.  
Elige la versión de 32 bits de instalación offline, *Windows x86 Offline*, y descarga el fichero **jdk-7u21-windows-i586.exe**.
- 1.3. Instala la versión de *JDK* descargada previamente.

A la hora de instalar *Tomcat* tenemos dos opciones, utilizar un instalador de *Windows* que realiza de forma automática la instalación creando un servicio asociado a *Tomcat* o realizar manualmente la instalación. Vamos a ver ambas aunque la segunda nos dará más posibilidades de personalización.

### 2. Instalación de *Tomcat* utilizando el instalador de *Windows*.

- 2.1. Descarga de <http://tomcat.apache.org/> la versión **32-bit/64-bit Windows Service Installer**.
- 2.2. Instala la versión descargada anteriormente, marcando la instalación completa y dejando el resto de opciones por defecto, Figuras 7.1, 7.2:

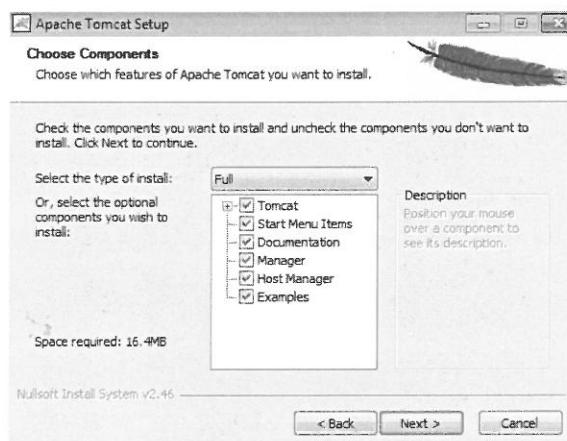


Figura 7.1: Instalación automática

- 2.3. Esta instalación automatizada ha creado un servicio asociado a *Tomcat* que se iniciará al arrancar *Windows*.
- 2.4. Comprueba que el servidor está iniciado, en la barra de tareas abajo a la izquierda hay un ícono que muestra una herramienta para monitorizar el servidor, Figura 7.3.

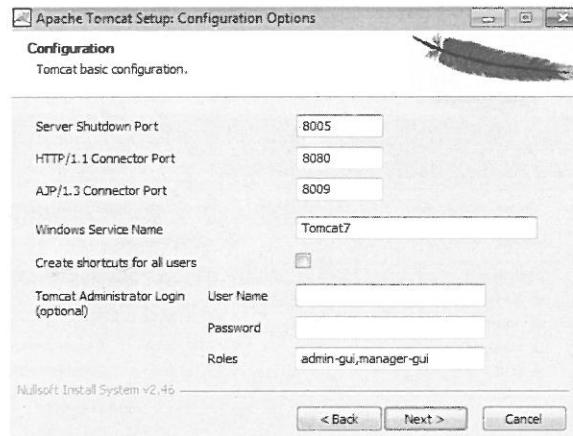


Figura 7.2: Instalación automática

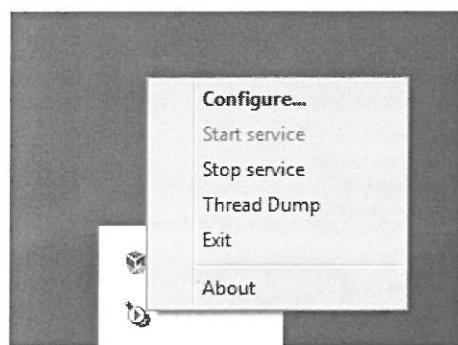


Figura 7.3: Consola de administración

2.5. Comprueba que el servidor está escuchando en el puerto 8080/TCP.

```
netstat -a -p TCP -n
```

2.6. Consulta el contenido del directorio C:\Program Files\Apache Software Foundation\Tomcat 7.0\webapps, Figura 7.4.

2.7. Consulta el contenido del directorio C:\Program Files\Apache Software Foundation\Tomcat 7.0\webapps\ROOT.

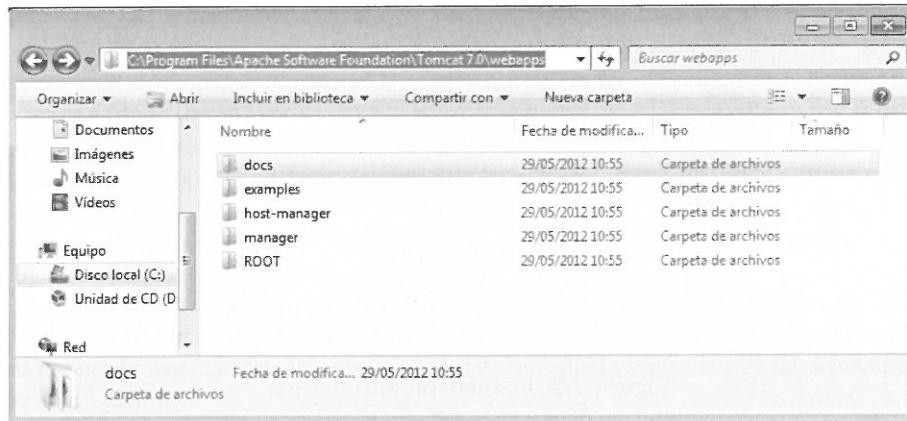


Figura 7.4: Directorio C:\Program Files\Apache Software Foundation\Tomcat 7.0\webapps

2.8. Para comprobar el correcto funcionamiento de *Tomcat* accede con el navegador a la dirección <http://localhost:8080>, Figura 7.5.

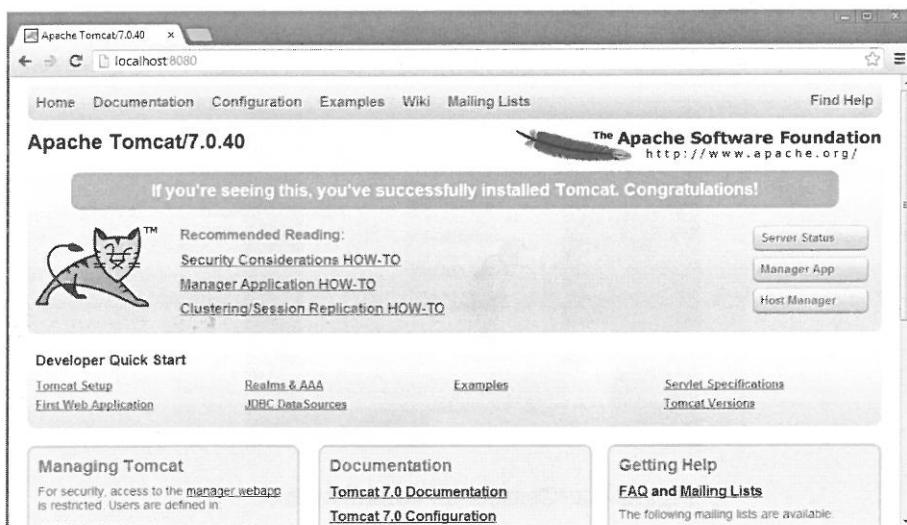
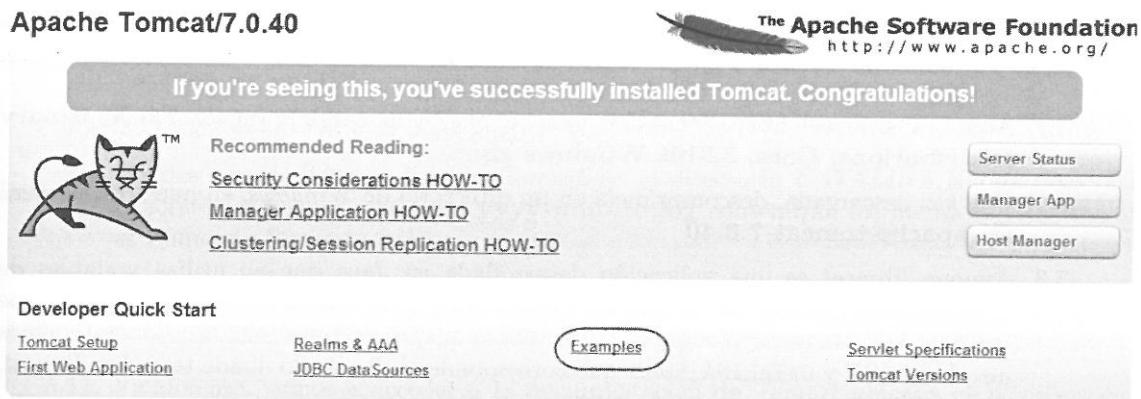


Figura 7.5: Comprobación del funcionamiento

### 3. Ejecución de *Servlets* en *Tomcat*.

- 3.1. Para terminar de verificar el correcto funcionamiento de la instalación vamos a ejecutar algún *servlet* de los que vienen incluidos con la instalación de *Tomcat*.
- 3.2. Arranca el servidor y accede al mismo. En la página de inicio selecciona **Examples**, Figura 7.6.

Apache Tomcat/7.0.40

Figura 7.6: Ejemplos *servlets*

- 3.3. Ejecuta el *servlet* *HelloWorld*, Figura 7.7.

### Servlet Examples with Code

This is a collection of examples which demonstrate some of the more frequently used parts of the Servlet API. Familiarity with the Java(tm) Programming Language is required. These examples will only work when viewed via an http URL. They will not work if you are viewing these pages via a "file://" URL. Please refer to the README file for instructions on how to configure and start the provided web server.

Wherever you see a form, enter some data and see how the servlet reacts. When playing with the Cookie and Session Examples, jump back to the Headers JSP page for more information.

To navigate your way through the examples, the following icons will help:

- Execute the example
- Look at the source code for the example
- Return to this screen

Tip: To see the cookie interactions with your browser, try turning on the "notify when setting a cookie" option in your browser preferences. This will let you see what cookies are being set by the examples.

Hello World

Figura 7.7: Ejemplos *servlets*

#### 4. Desinstalar Tomcat

- 4.1. Para el servicio asociado a Tomcat.
- 4.2. Accede a **Inicio, Todos los programas, Apache Tomcat 7.0 Tomcat 7, Unistall Tomcat 7.0** y sigue los pasos del asistente.

#### 5. Instalación manual de Tomcat.

Para gestionar todos los aspectos de la instalación de Tomcat, gestionar las variables de entorno y administrar Tomcat desde línea de comandos vamos a instalar Tomcat manualmente.

- 5.1. Accede al sitio web <http://tomcat.apache.org/> y descarga la versión **7.0.X, Binary Distributions, Core, 32-bit Windows zip**.
- 5.2. Una vez descargada, descomprímela en un directorio de Windows, en nuestro caso será **C:\apache-tomcat-7.0.40**
- 5.3. Aunque Tomcat es una aplicación desarrollada en Java que no utiliza variables de entorno, para ejecutar los *scripts* de administración necesitamos crear dos variables de entorno de sistema en Windows, **JAVA\_HOME** que apuntará al directorio donde tenemos instalado *JRE* y **CATALINA\_HOME** que corresponde al directorio donde tenemos Tomcat,

Figura 7.8.

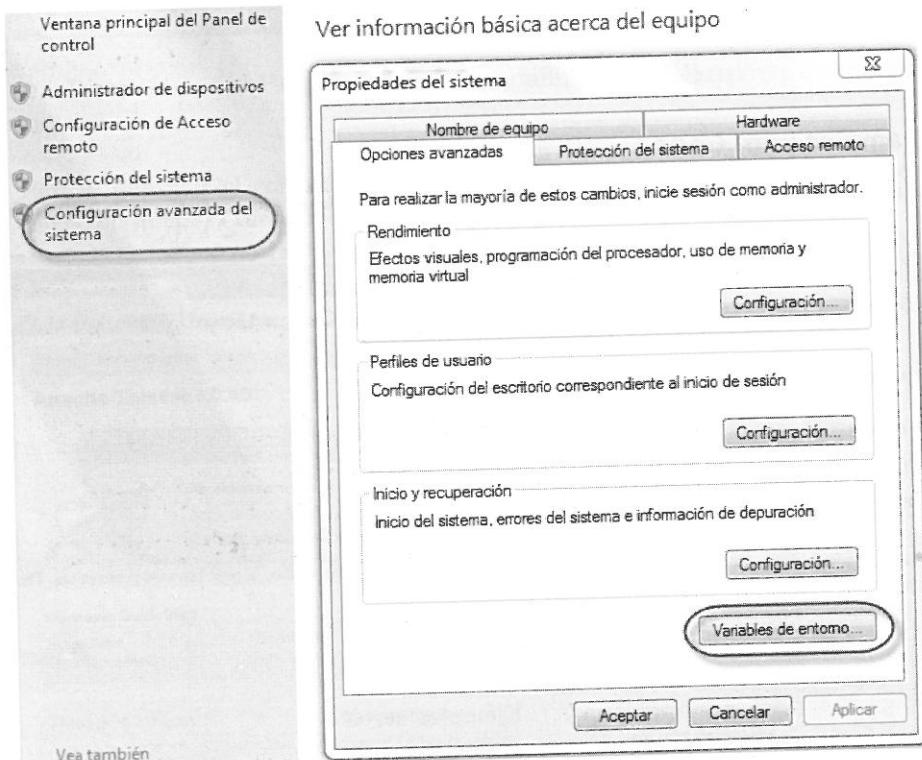


Figura 7.8: Creación de variables de entorno

- 5.4. Para comprobar que las variables de entorno han sido creadas correctamente ejecuta los comandos:

```
echo %CATALINA_HOME%
echo %JAVA_HOME%
```

- 5.5. Si no aparecieran reinicia la máquina.  
 5.6. Para comprobar el funcionamiento de *Tomcat*, inicia el servidor ejecutando el *script*:

```
%CATALINA_HOME%\bin\startup.bat (C:\apache-tomcat-7.0.40\bin\startup.bat)
```

- 5.7. Comprueba que el servidor está escuchado en el puerto **TCP 8080**:

```
netstat -a -p TCP -n
```

- 5.8. Accede a <http://localhost:8080>.  
 5.9. Puedes ver los ficheros *logs* de *Tomcat* en el directorio *CATALINA\_HOME/logs*. En concreto los ficheros **catalina.yyyy-mm-dd.log** contendrán los mensajes relativos al arranque de *Tomcat*.

- 5.10. Para parar *Tomcat* ejecuta el script:

```
%CATALINA_HOME%\bin\shutdown.bat (C:\apache-tomcat-7.0.40\bin\shutdown.bat)
```

- 5.11. Finalmente, vamos a acceder a la documentación de *Tomcat* ubicada en el directorio **CATALINA\_HOME\webapps\docs\index.html**, Figura 7.9.



**Documentation Index**

**Introduction**

This is the top-level entry point of the documentation bundle for the Apache Tomcat Servlet/JSP container. Apache Tomcat version 7.0 implements the Servlet 3.0 and JavaServer Pages 2.2 specifications from the [Java Community Process](#), and includes many additional features that make it a useful platform for developing and deploying web applications and web services.

Select one of the links from the navigation menu (to the left) to drill down to the more detailed documentation that is available. Each available manual is described in more detail below.

**Apache Tomcat User Guide**

The following documents will assist you in downloading, installing Apache Tomcat 7, and using many of the Apache Tomcat features.

- 1. [Introduction](#) - A brief, high level, overview of Apache Tomcat.
- 2. [Setup](#) - How to install and run Apache Tomcat on a variety of platforms.
- 3. [First web application](#) - An introduction to the concepts of a *web application* as defined in the Servlet Specification. Covers basic organization of your web application source tree, the structure of a web application archive, and an introduction to the web application deployment descriptor (*/WEB-INF/web.xml*).
- 4. [Deployer](#) - Operating the Apache Tomcat Deployer to deploy, precompile, and validate web applications.
- 5. [Manager](#) - Operating the Manager web app to deploy, undeploy, and redeploy applications while Apache Tomcat is running.
- 6. [Realms and Access Control](#) - Description of how to configure *Realms* (databases of users, passwords, and their associated roles) for use in web applications that utilize Container Managed Security.
- 7. [Security Manager](#) - Configuring and using a Java Security Manager to support fine-grained control over the behavior of your web applications.
- 8. [JNDI Resources](#) - Configuring standard and custom resources in the JNDI naming context that is provided to each web application.

Figura 7.9: Documentación *Tomcat*

## 6. Ejecución de *Servlets* en *Tomcat*.

- 6.1. Para terminar de verificar el correcto funcionamiento de la instalación vamos a ejecutar algún *servlet* de los que vienen incluidos con la instalación de *Tomcat*.  
 6.2. Arranca el servidor y accede al mismo. En la página de inicio selecciona **Examples**.  
 6.3. Ejecuta el *servlet* *HelloWorld*.

## 7. Errores típicos en la instalación de *Tomcat*.

- 7.1. El error más común viene dado por tener instalado en la misma máquina un servidor web que esté escuchando en el puerto **8080**. Este es el puerto por defecto que *Tomcat* utiliza para escuchar las peticiones y lo hace en este puerto para diferenciarlo del puerto **80**, que es el habitualmente empleado por los servidores Web. Para cambiar el puerto por el que escucha peticiones *Tomcat* hay que editar el fichero de configuración **CATALINA\_HOME\conf\server.xml**, cambiarlo por un puerto que no esté en uso y reiniciar *Tomcat*.
- 7.2. Otros errores que se puede producir pueden ser debidos a la presencia de un servidor *proxy*. Si estamos accediendo de forma local, debemos indicar en la configuración del navegador que no utilice el servidor *proxy* para las direcciones locales.
- 7.3. Si la ejecución del *servlet* no funcionara deberíamos comprobar que la variable de entorno **JAVA\_HOME** apunta al directorio donde está instalado *JRE*.
- 7.4. Si realizamos la instalación automática de *Tomcat* al reiniciar el equipo puede que obtengamos el mensaje de error: “acceso denegado unable to open the service tomcat 7”. Para solucionarlo debemos ir al fichero **C:\Program Files\Apache Software Foundation\Tomcat 7.0\bin\Tomcat7w** y marcar la casilla **Ejecutar este programa como administrador**.

## **7.2. Instalación y configuración básica de *Eclipse***

En esta práctica se instalará la herramienta de desarrollo *Eclipse* en la máquina virtual **DesarrolloW7XX**. Como en dicho equipo tenemos instalado el servidor de aplicaciones *Tomcat 7*, resultará sencillo probar los diferentes desarrollos que vayamos realizando.

*Eclipse* es un entorno de desarrollo de *software* compuesto por un *IDE* y por un conjunto de *plugins* que permiten extender su funcionalidad. Es un proyecto *opensource* controlado por la *Eclipse foundation, eclipse.org*. Aunque inicialmente fue lanzado por *IBM* en la actualidad es patrocinado por diferentes empresas de *software*. *Eclipse* provee herramientas que facilitan el desarrollo de aplicaciones en *Java*, lo que constituye su principal utilidad aunque no la única.

Como *Eclipse* es una aplicación basada en *Java*, para su funcionamiento es necesario tener instalado la máquina virtual de *Java (JRE)*.

Además, para desarrollar *JSPs* y *Servlets* usaremos las librerías respectivas, contenidas en la carpeta **lib** de la instalación de *Tomcat*.

### 1. Instalación de *Eclipse*.

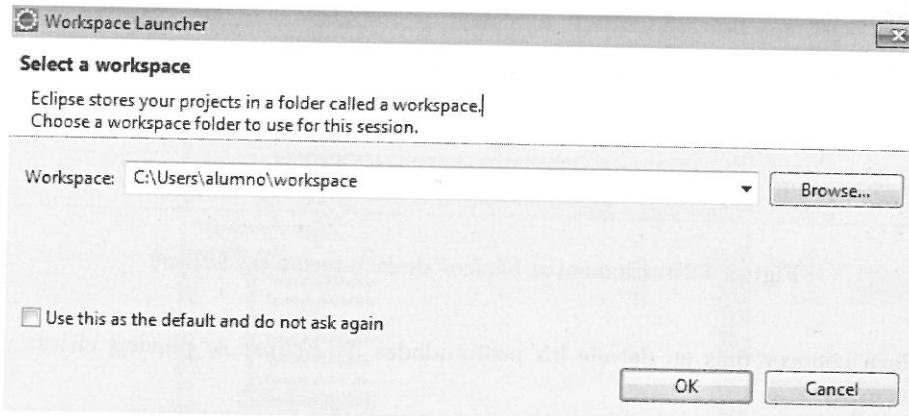
- 1.1. Inicia sesión en el equipo **DesarrolloW7XX**, accede a la página de descargas de *Eclipse*, <http://www.eclipse.org/downloads/>, y elige *Eclipse IDE for Java EE Developers*, Figura 7.10. Esta versión de *Eclipse* nos facilitará el despliegue de aplicaciones en el servidor *Tomcat*. Frente a otras versiones, la denominada *Eclipse IDE for Java EE Developers* incluye integración con los servidores de aplicaciones más frecuentes y diferentes herramientas para el desarrollo de aplicaciones *Java Enterprise Edition*, en el siguiente enlace puedes ver una comparativa de las funcionalidades de cada versión: <http://www.eclipse.org/downloads/compare.php>.

Figura 7.10: *Eclipse IDE for Java EE Developers*

- 1.2. Descarga el fichero .zip en tu equipo y descomprímelo. Como *Eclipse* es una aplicación portable que no necesita instalación, simplemente crea un acceso directo en el escritorio al fichero *eclipse.exe*.

## 2. Configuración básica de *Eclipse*.

- 2.1. Al arrancar *Eclipse*, *eclipse.exe*, deberemos elegir un directorio como espacio de trabajo, *workspace*, Figura 7.11. En este directorio se guardarán opciones de configuración de *Eclipse* y los diferentes proyectos que vayamos desarrollando.

Figura 7.11: *Workspace*

- 2.2. Una vez iniciada la aplicación obtendremos información sobre diferentes tutoriales relativos a la herramienta, Figura 7.12.
- 2.3. Una vez cerrada la ayuda accedemos al entorno de desarrollo, dividido en diferentes partes según la perspectiva que elijamos. Por defecto está activada la perspectiva *Java EE*, Figura 7.13.

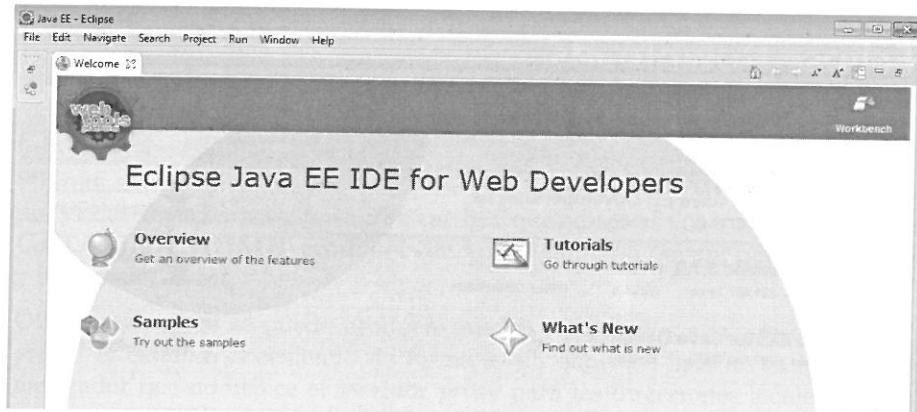


Figura 7.12: Inicio de *Eclipse*

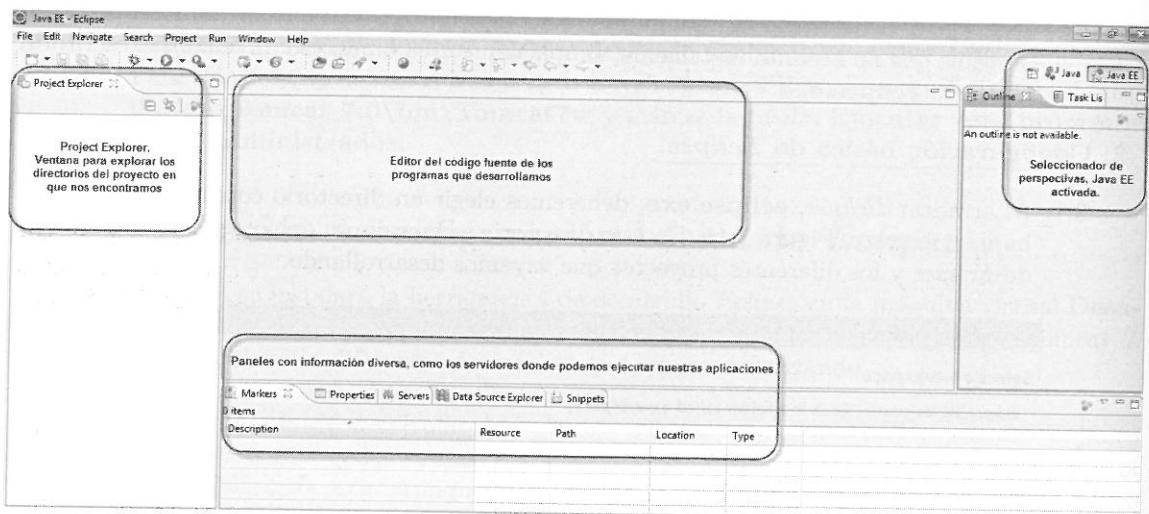


Figura 7.13: Elementos básicos de la interfaz de *Eclipse*

2.4. Para conocer más en detalle las posibilidades de *Eclipse* se pueden visitar diferentes recursos:

- <http://www.eclipse.org/resources/>
- <http://www.eclipse.org/documentation/>
- <http://eclipsutorial.sourceforge.net/>

## 7.3. Integración de *Eclipse* con *Tomcat*

En esta práctica integraremos *Eclipse* con *Tomcat* de forma que podamos probar nuestros desarrollos directamente en el servidor de aplicaciones.

### 1. Registro del servidor *Tomcat* en *Eclipse*.

1.1. Inicia sesión en el equipo DesarrolloW7XX con el usuario **alumno** e inicia *Eclipse*.

Estará configurado con la perspectiva *Java EE*. Si no fuera así, puedes cambiarla en la opción de menú **Windows, Open Perspective**.

- 1.2. En el menú **Windows, Show View**, accede a la opción **Servers**, Figura 7.14.

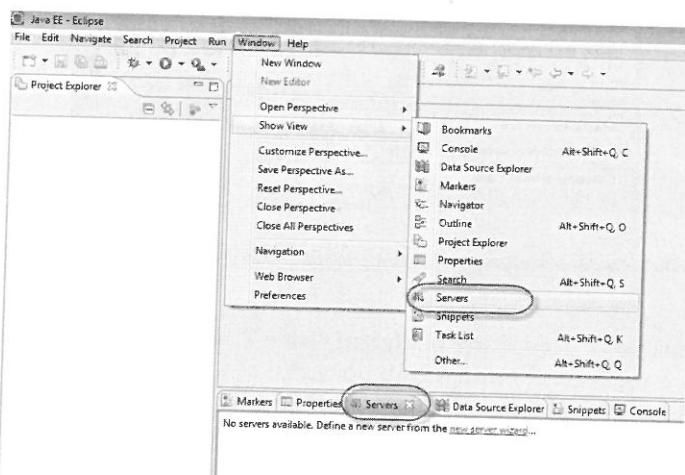


Figura 7.14: Añadir servidor de aplicaciones a *Eclipse*

- 1.3. Pulsa en **new server wizard** o botón derecho del ratón, **new, server** y elige el servidor de aplicaciones que tenemos instalado, **Apache Tomcat 7**, Figura 7.15. Observa cómo *Eclipse* puede integrarse con otros servidores de aplicaciones como *WebSphere*, *JBOSS*, etc.

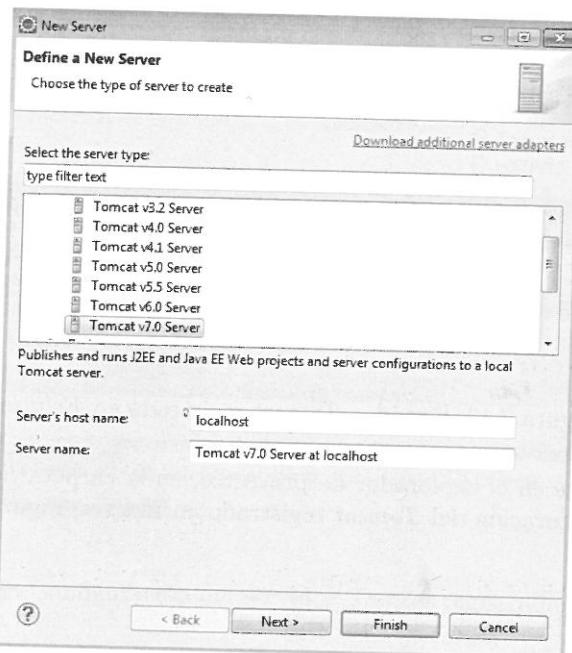


Figura 7.15: Añadir servidor de aplicaciones a *Eclipse*

- 1.4. Especifica el nombre que quieras dar al servidor, el directorio donde está instalado, la máquina virtual de Java y pulsa en **Finalizar**, Figura 7.16.

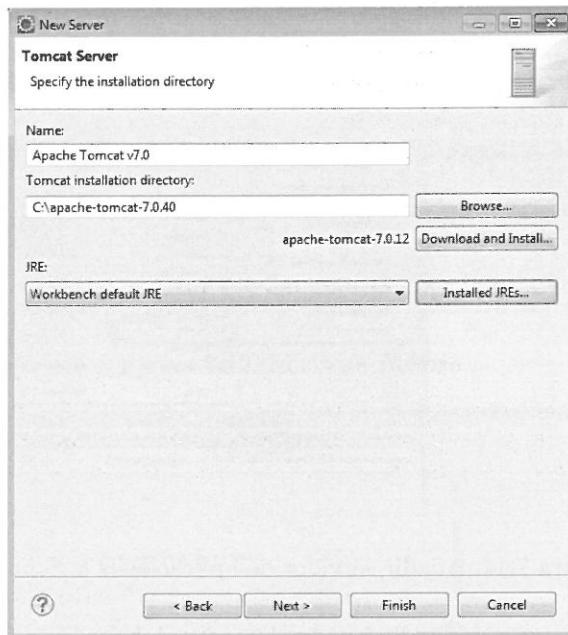


Figura 7.16: Añadir servidor de aplicaciones a *Eclipse*

## 2. Relación entre el Servidor *Tomcat* registrado en *Eclipse* y la instalación de *Tomcat*.

Una vez registrado el servidor *Tomcat* en *Eclipse* veremos que aparece en la pestaña *Servers*, Figura 7.17.



Figura 7.17: Servidor *Tomcat* registrado en *Eclipse*

También nos aparece en el explorador de proyectos, en la carpeta *Servers*, una carpeta con los ficheros de configuración del *Tomcat* registrado en *Eclipse*, Figura 7.18.

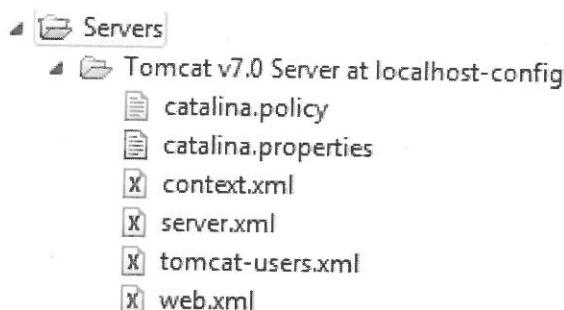


Figura 7.18: Ficheros de configuración del servidor *Tomcat* registrado en *Eclipse*

Esto quiere decir que el servidor *Tomcat* instalado en *Eclipse* tiene sus propios ficheros de configuración diferentes de los de la instalación de *Tomcat*. Si hacemos doble clic sobre el servidor *Tomcat* en la pestaña *Servers* podemos comprobar cómo la configuración se guarda en la *metadata* del *workspace* de *Eclipse* y no se modifica la instalación original de *Eclipse*, Figura 7.19.

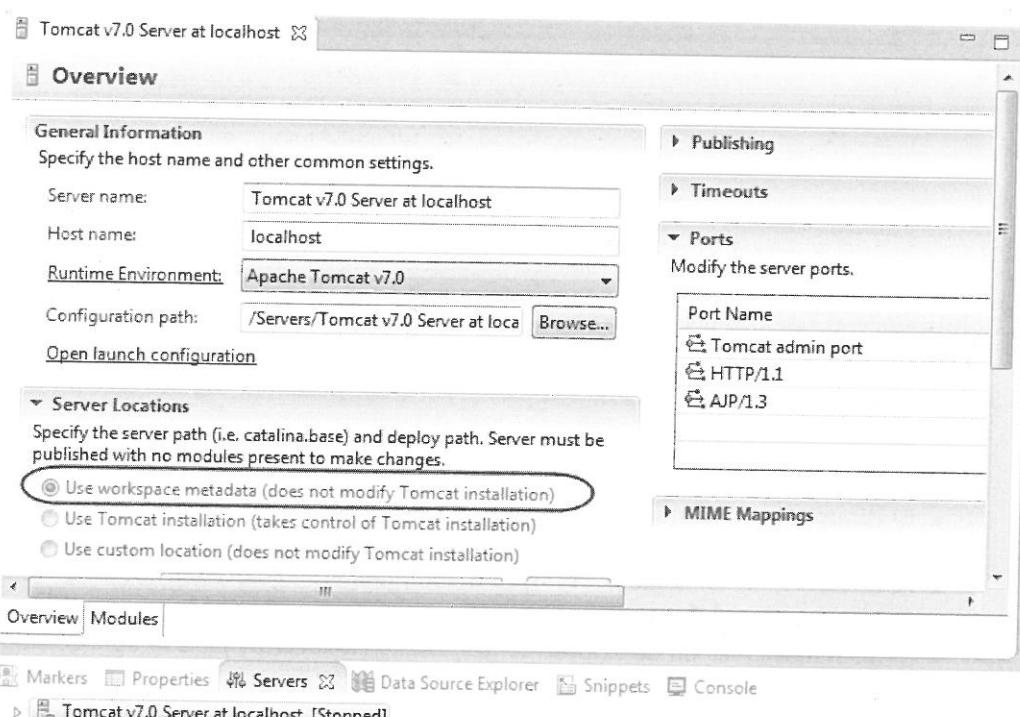


Figura 7.19: Configuración del servidor *Tomcat* registrado en *Eclipse*

Se puede comprobar también cómo *Tomcat* está preparado para republicar automáticamente los proyectos cuando se realicen cambios en los ficheros fuente, Figura 7.20.

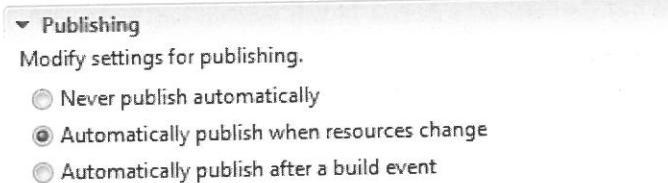


Figura 7.20: Republicación automática

### 3. Prueba de la integración entre *Eclipse* y *Tomcat*.

Para comprobar la integración entre *Eclipse* y *Tomcat* vamos a realizar un primer proyecto.

- 3.1. Crea un proyecto web dinámico pulsando en **File**, **New**, **Dynamic Web Project**, Figura 7.21.

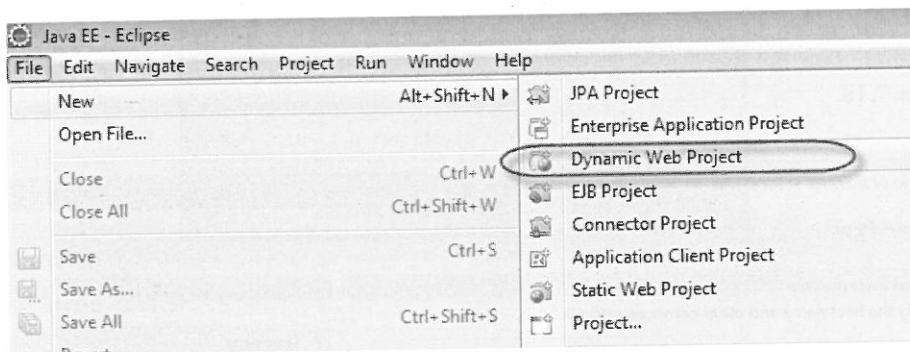


Figura 7.21: Creación de un proyecto web dinámico

- 3.2. Introduce el nombre del proyecto, en nuestro caso **PrimerProyecto**, y pulsa en **Siguiente**, Figura 7.22.

- 3.3. Las dos siguientes ventanas, Figuras 7.23 y 7.24, nos permiten establecer:

- El directorio donde estarán las fuentes de *Java* (**src**).
- El directorio donde se guardarán los archivos precompilados **build\classes**.
- El directorio por el que empezarán todas las URLs de los recursos de nuestro proyecto (**Context root**), que por defecto *Eclipse* utiliza el nombre del proyecto.
- El directorio de contenidos (**WebContent**) donde se guardarán los archivos **html** y **jsp**.
- Si queremos crear un descriptor de despliegue (**web.xml**) para la aplicación.

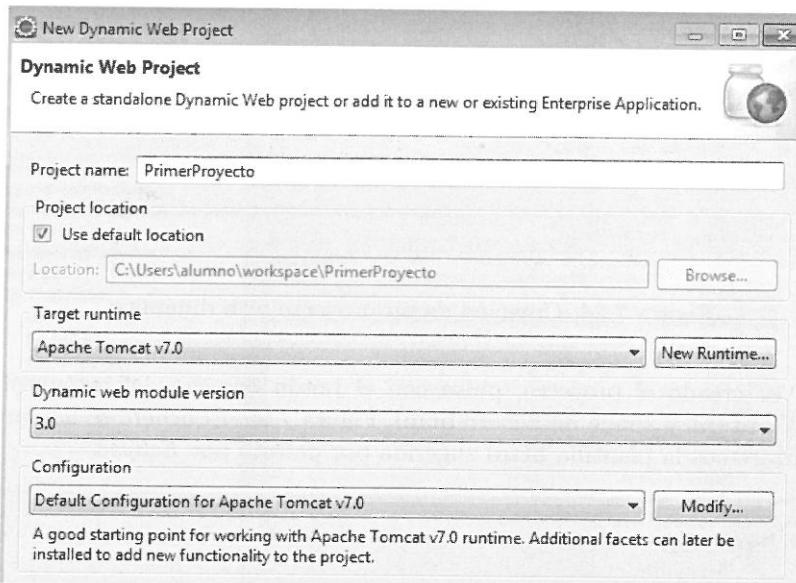


Figura 7.22: Creación de un proyecto web dinámico

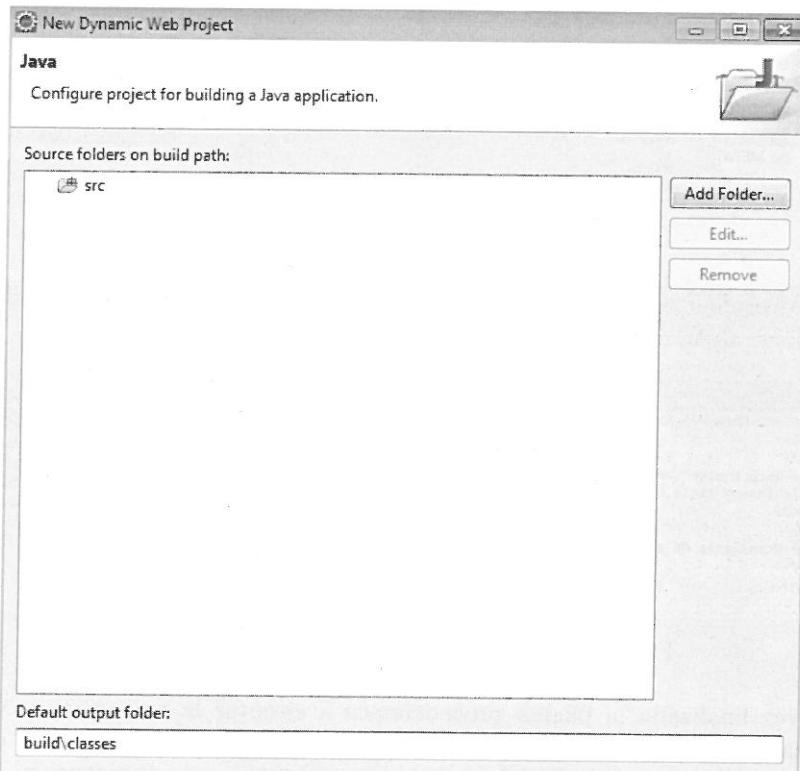


Figura 7.23: Creación de un proyecto web dinámico

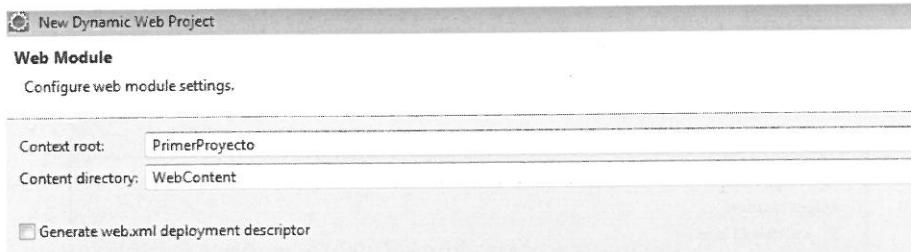


Figura 7.24: Creación de un proyecto web dinámico

- 3.4. Una vez creado el proyecto, pulsa con el botón derecho del ratón sobre la carpeta **WebContent** y añade un fichero **html**, Figura 7.25. El fichero se llamará **index.html** y utilizaremos la plantilla **html** sugerida por defecto por *Eclipse*.

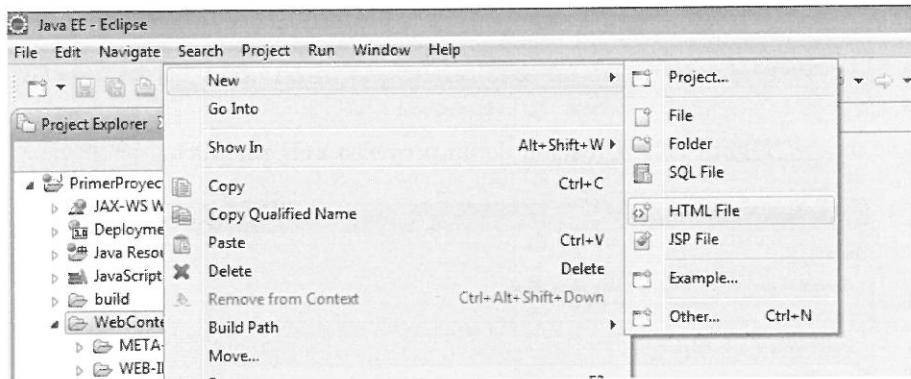


Figura 7.25: Añadir fichero index.html

- 3.5. A continuación añade algún texto dentro del cuerpo del fichero **index.html** para poder visualizar algún contenido en la página, Figura 7.26.

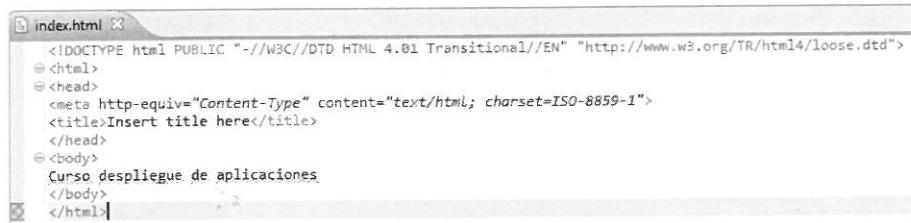


Figura 7.26: Edición del fichero index.html

- 3.6. Una vez finalizada la página procederemos a ejecutar la aplicación sobre el servidor Tomcat que definimos en el primer apartado. Para ello seleccionaremos el proyecto en la ventana **Project Explorer** y pulsaremos el botón verde de ejecución. Nos aparecerá un asistente para elegir el servidor donde se ejecutará la aplicación, elige el definido en el apartado inicial de la práctica, Figura 7.27.
- 3.7. Es importante que antes de iniciar *Tomcat* comprobemos que el servidor no haya sido arrancado previamente.

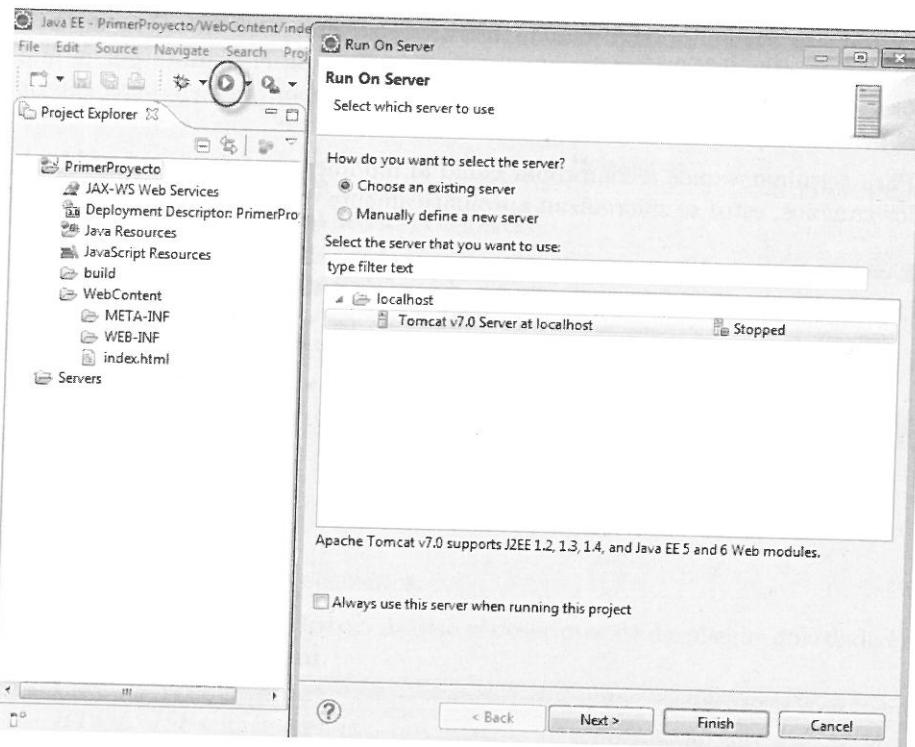


Figura 7.27: Ejecución del proyecto

- 3.8. Al terminar el asistente y pulsar en **Finalizar** nuestro proyecto se ejecutará en el servidor *Tomcat*, Figura 7.28.

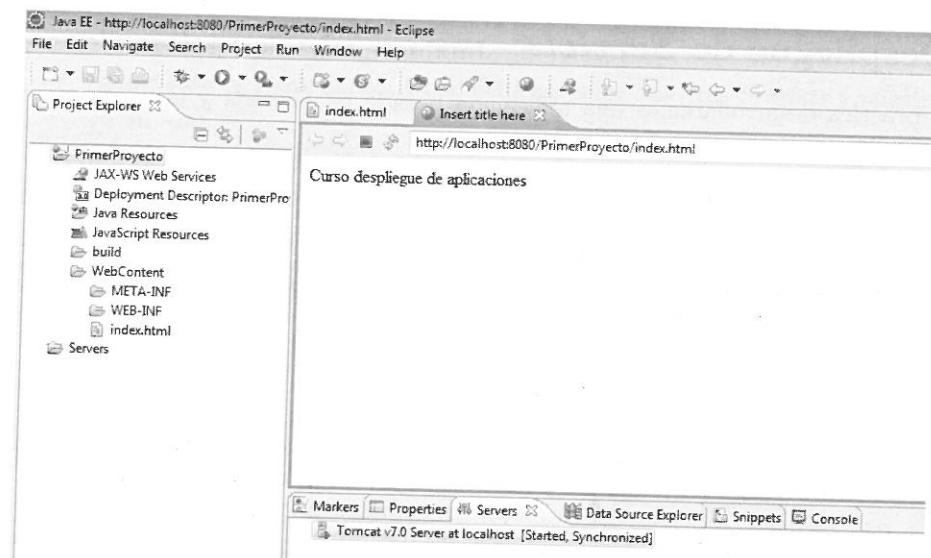


Figura 7.28: Ejecución del proyecto

- 3.9. Observa la dirección `http://localhost:8080/PrimerProyecto/index.html`, aparece el nombre del equipo y el puerto que utiliza *Tomcat*, el nombre por el que empiezan todos los recursos de nuestro proyecto (el *Context root* que definimos anteriormente) y el fichero al que estamos accediendo.
- 3.10. Para terminar vamos a comprobar cómo al modificar el fichero `index.html` y guardar los cambios, estos se sincronizan automáticamente con el servidor *Tomcat*, Figura 7.29.

The screenshot shows the Eclipse IDE interface. In the center, there is a code editor window titled "index.html" containing the following HTML code:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
    <title>Insert title here</title>
  </head>
  <body>
    Curso despliegue de aplicaciones modificado
  </body>
</html>
```

Below the code editor, the Eclipse interface includes several toolbars and a status bar. At the bottom, the "Servers" view is active, showing a single entry:

- Tomcat v7.0 Server at localhost [Started, Republish]

Figura 7.29: Republicación del proyecto