

Contenido

Instalación del Java Development Kit (JDK)	3
Instalación de Apache Tomcat	8
Creación del certificado	14
Instalación de PostgreSQL.....	17
Integración CAS-Spring-proyecto-Base PostgreSQL.....	25

Instalación del Java Development Kit (JDK)

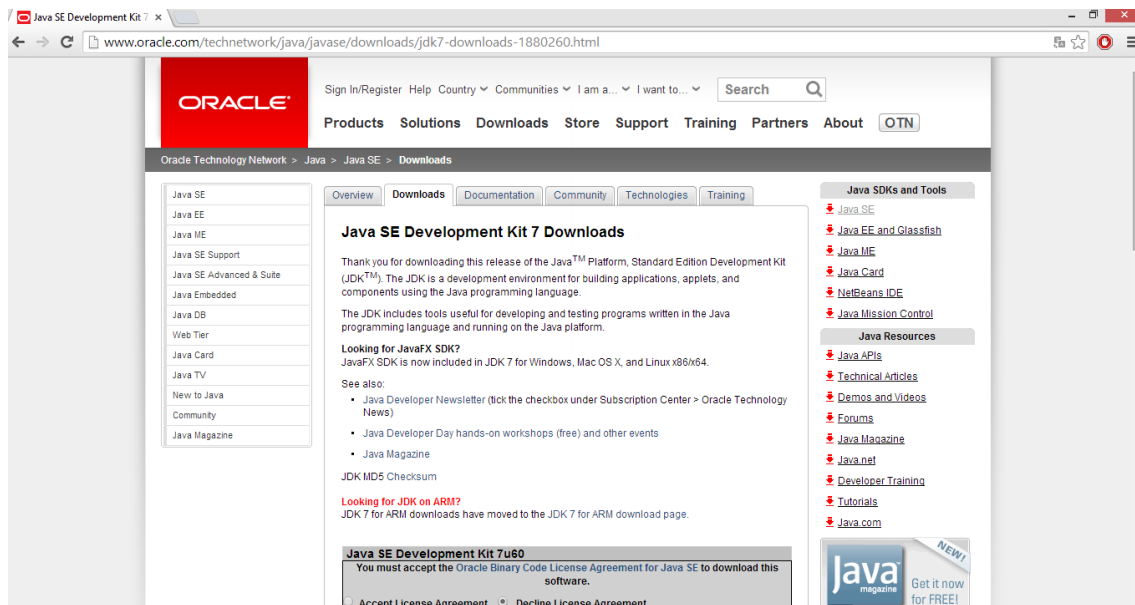
De gran importancia es que tengamos instalado el JDK, este debiera ser el primer paso a realizar. Puede sernos de gran ayuda el siguiente link

<http://www.luiskano.net/blog/2010/04/28/tutorial-instalar-java-jdk-en-windows-7/>

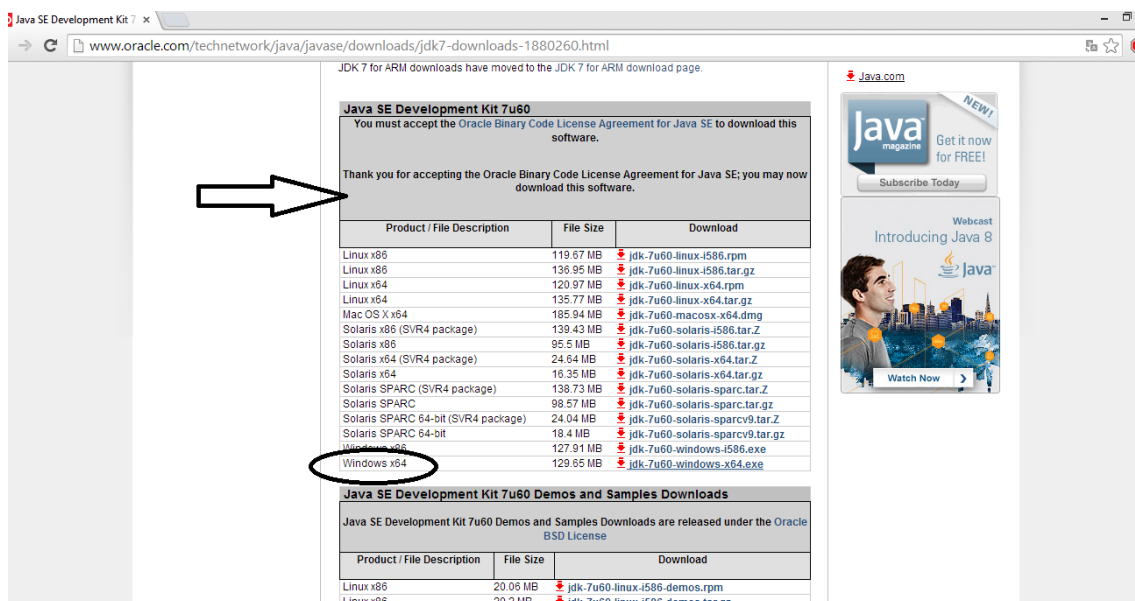
Que nos muestra la instalación para Windows 7, además que tiene otros enlaces para la instalación en otros sistemas de Windows.

La instalación que se mostrará aquí es de la versión jdk 7, la más actual al día 7 de julio del 2014.

1.- Nos dirigimos a la página de Oracle a la sección de descargas del jdk, eventualmente encontraremos la versión más actual a la fecha.



2.-Aceptamos las condiciones de la licencia para el JDK y escogemos la versión que corresponde a nuestro sistema operativo.



3.- La descarga inicia inmediatamente después, solo queda esperar a que termine.

Solaris SPARC	98.57 MB	jdk-7u60
Solaris SPARC 64-bit (SVR4 package)	24.04 MB	jdk-7u60
Solaris SPARC 64-bit	18.4 MB	jdk-7u60
Windows x86	127.91 MB	jdk-7u60
Windows x64	129.65 MB	jdk-7u60

Java SE Development Kit 7u60 Demos and Samples Downloads
Java SE Development Kit 7u60 Demos and Samples Downloads are under the [BSD License](#)

Product / File Description	File Size	Download
Linux x86	20.06 MB	jdk-7u60-linux-i586-d
Linux x86	20.2 MB	jdk-7u60-linux-i586-d

jdk-7u60-windows-x64.exe
26,8/130 MB, 7 minutos r...

4.- Comenzamos la instalación. Básicamente se instala todo por defecto, es decir, escogiendo siguiente en cada pantalla, sin mayor tipo de configuración.

Java SE Development Kit 7 Update 60 (64-bit) - Setup



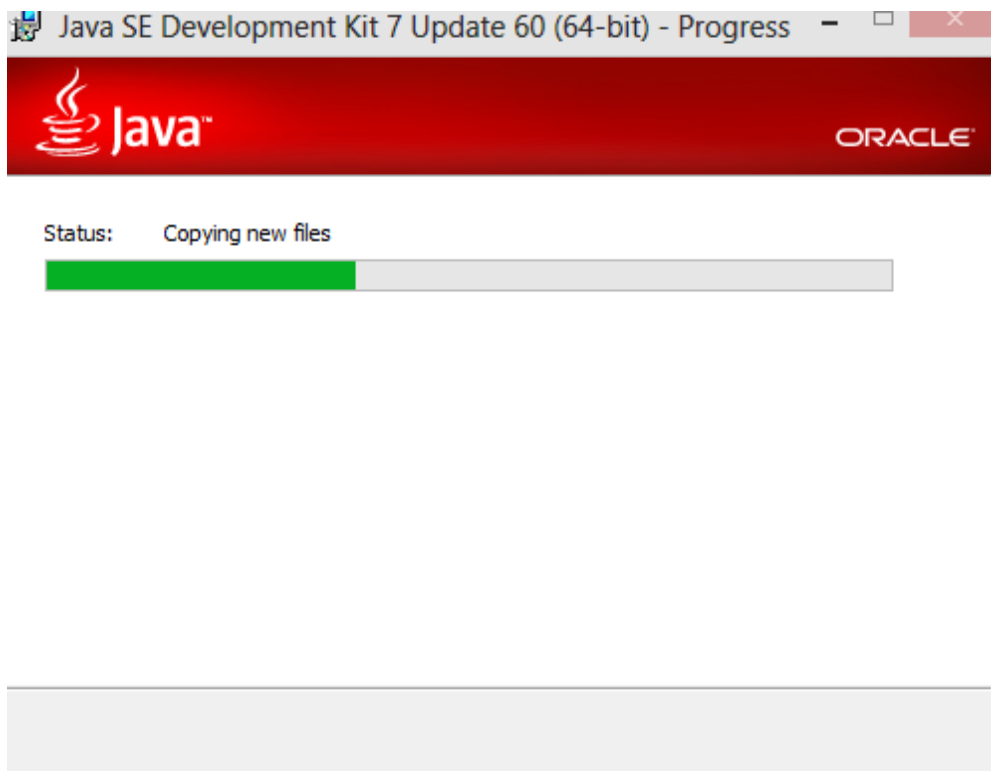
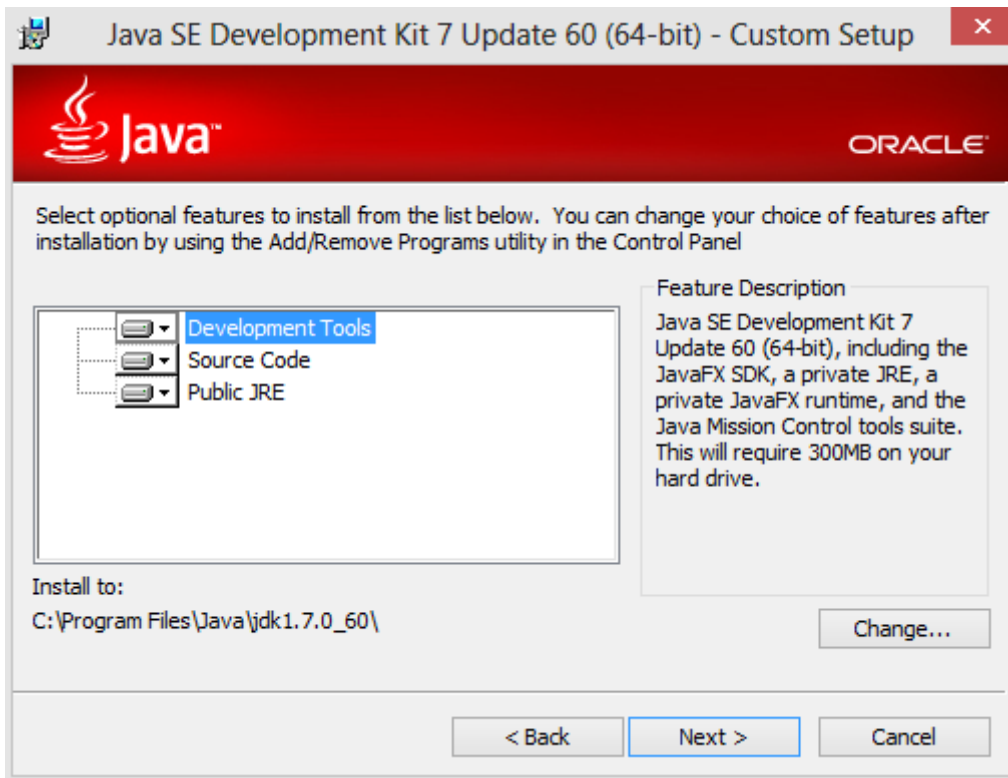
Welcome to the Installation Wizard for Java SE Development Kit 7 Update 60

This wizard will guide you through the installation process for the Java SE Development Kit 7 Update 60.

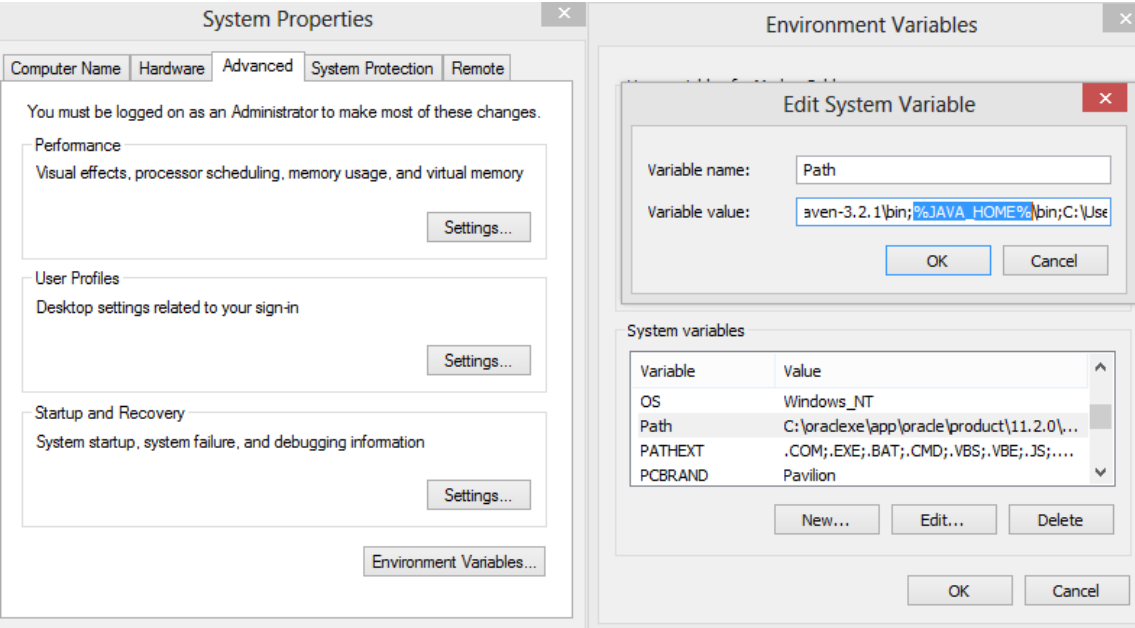
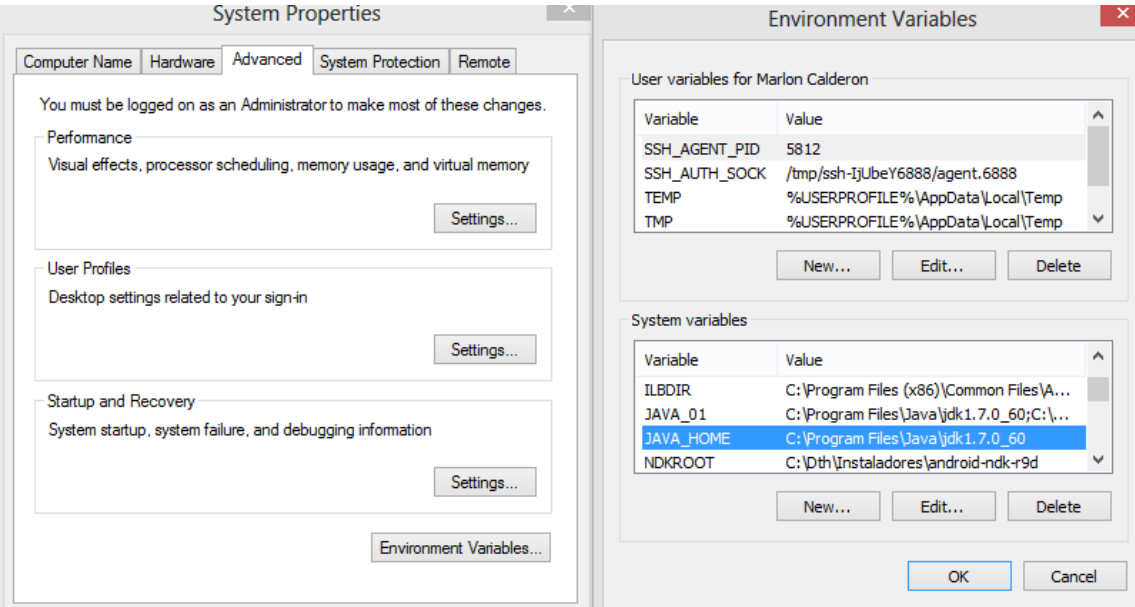
The Java Mission Control profiling and diagnostics tools suite is now available as part of the JDK.

Next >

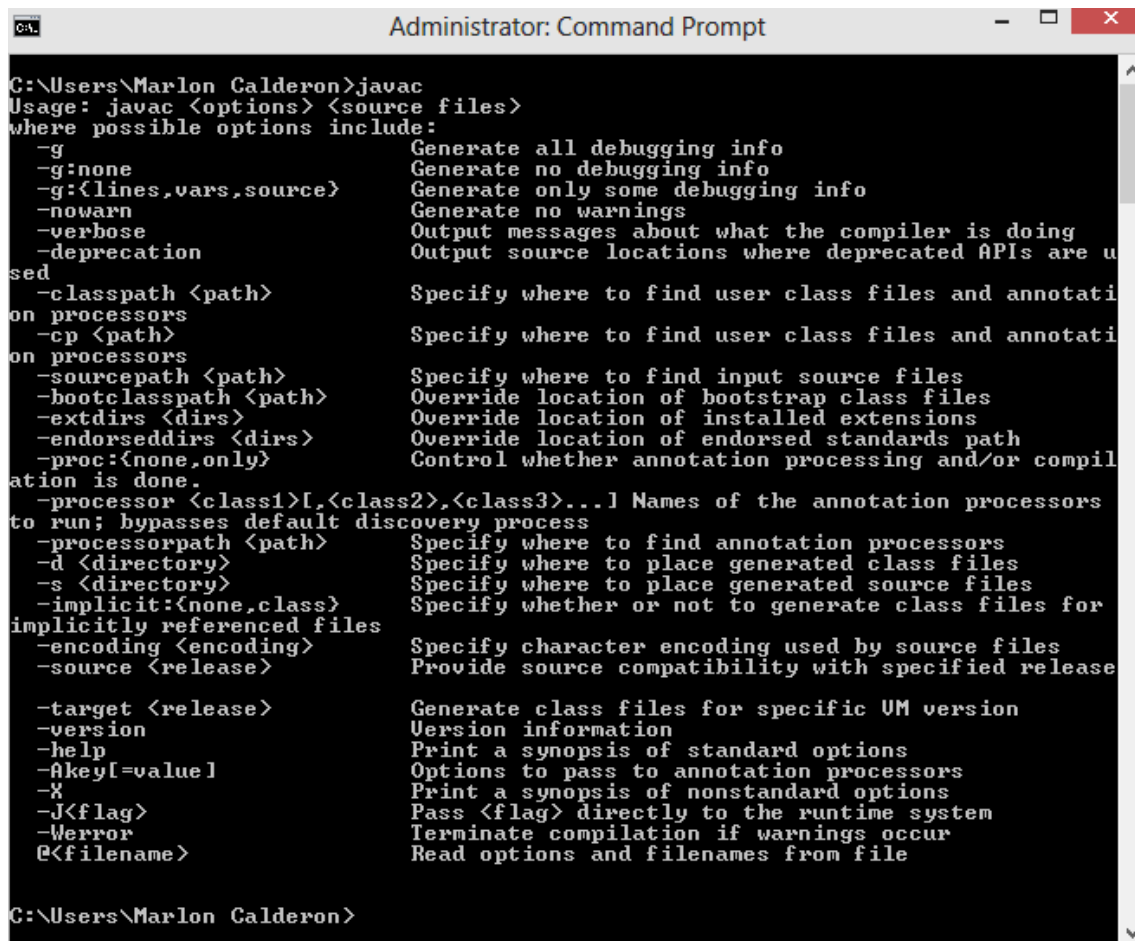
Cancel



5.- Se configura las variables de entorno



6.- Finalmente se verifica que está bien instalado



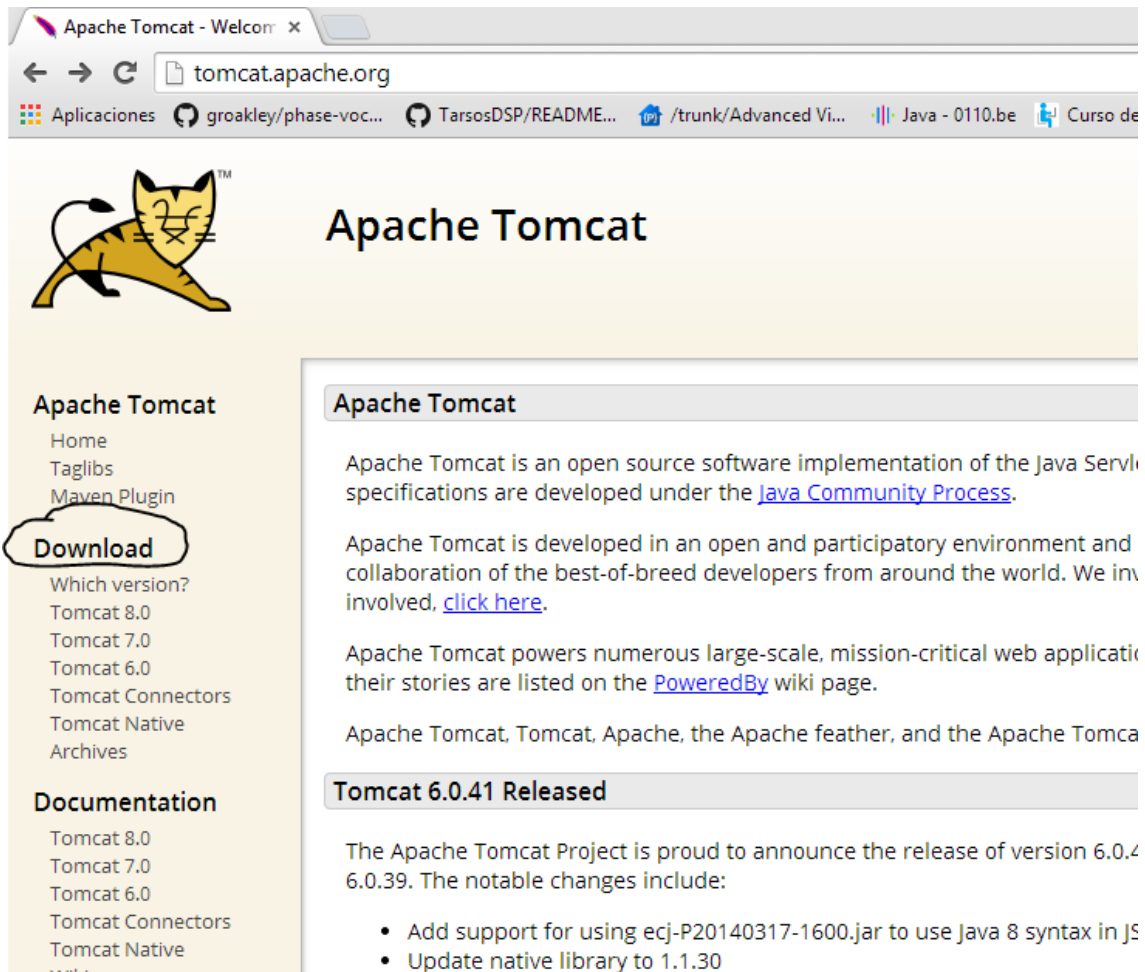
```
C:\Users\Marlon Calderon>javac
Usage: javac <options> <source files>
where possible options include:
  -g               Generate all debugging info
  -g:none          Generate no debugging info
  -g:{lines,vars,source}  Generate only some debugging info
  -nowarn          Generate no warnings
  -verbose         Output messages about what the compiler is doing
  -deprecation     Output source locations where deprecated APIs are used
  -classpath <path> Specify where to find user class files and annotations
  -processorpath <path> Specify where to find user class files and annotations
  -sourcepath <path> Specify where to find input source files
  -bootclasspath <path> Override location of bootstrap class files
  -extdirs <dirs> Override location of installed extensions
  -endorseddirs <dirs> Override location of endorsed standards path
  -proc:{none,only} Control whether annotation processing and/or compilation is done.
  -processor <class1>[,<class2>,<class3>...] Names of the annotation processors to run; bypasses default discovery process
  -processorpath <path> Specify where to find annotation processors
  -d <directory> Specify where to place generated class files
  -s <directory> Specify where to place generated source files
  -implicit:{none,class} Specify whether or not to generate class files for implicitly referenced files
  -encoding <encoding> Specify character encoding used by source files
  -source <release> Provide source compatibility with specified release
  -target <release> Generate class files for specific VM version
  -version         Version information
  -help            Print a synopsis of standard options
  -Akey[=value]    Options to pass to annotation processors
  -X              Print a synopsis of nonstandard options
  -J<flag>         Pass <flag> directly to the runtime system
  -Werror          Terminate compilation if warnings occur
  @<filename>      Read options and filenames from file

C:\Users\Marlon Calderon>
```

Instalación de Apache Tomcat

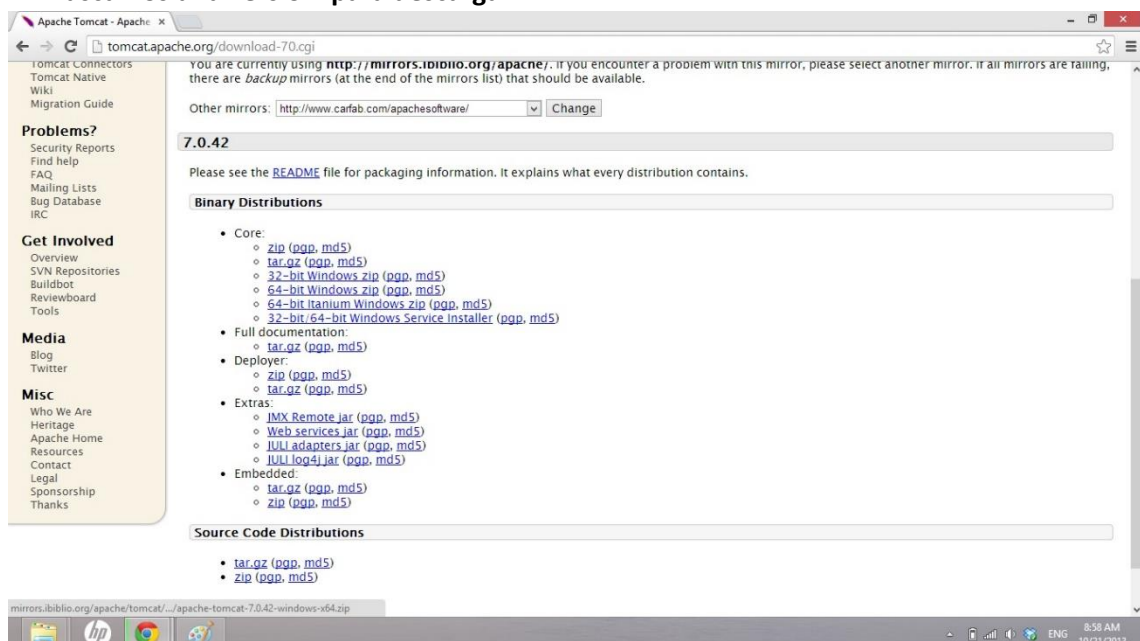
Instalación de Apache Tomcat 7.0.42

1.- Vamos directo a la página de apache a la sección de descargas



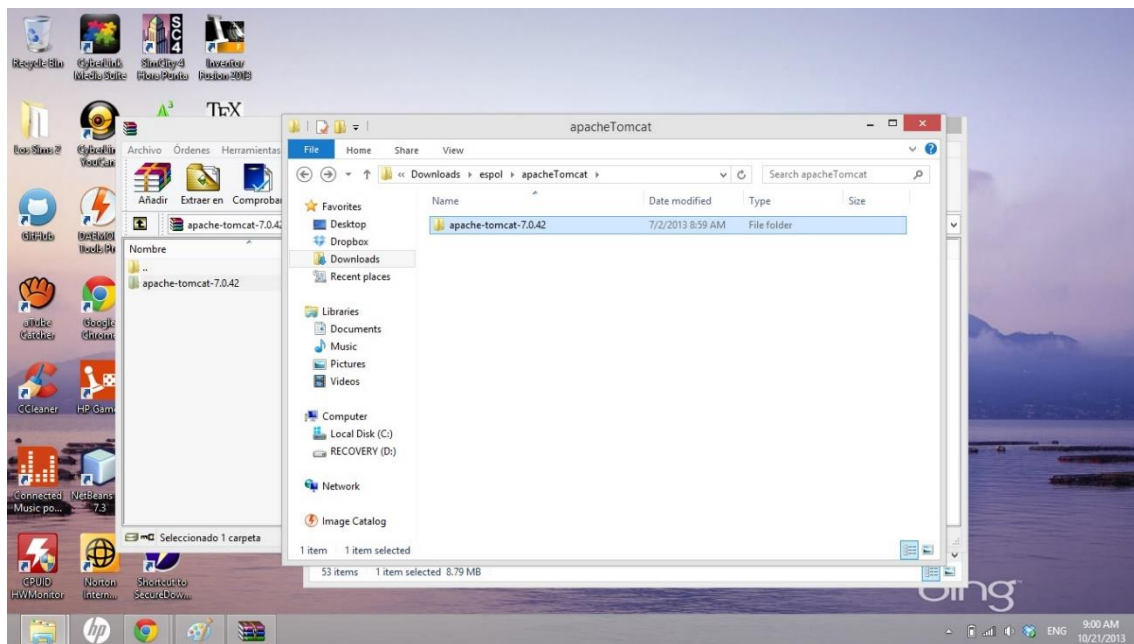
The screenshot shows the Apache Tomcat homepage in a web browser. The browser's address bar displays 'tomcat.apache.org'. On the left sidebar, the 'Download' link is circled in black. The main content area features the Apache Tomcat logo (a yellow cat) and the title 'Apache Tomcat'. Below the logo, there is a navigation menu with links: Home, Taglibs, Maven Plugin, Download (circled), Which version?, Tomcat 8.0, Tomcat 7.0, Tomcat 6.0, Tomcat Connectors, Tomcat Native, and Archives. The 'Download' section is expanded, showing links for 'Which version?', 'Tomcat 8.0', 'Tomcat 7.0', 'Tomcat 6.0', 'Tomcat Connectors', 'Tomcat Native', and 'Archives'. The 'Documentation' section is also visible, with links for 'Tomcat 8.0', 'Tomcat 7.0', 'Tomcat 6.0', 'Tomcat Connectors', and 'Tomcat Native'. The main content area contains a section titled 'Apache Tomcat' with a description: 'Apache Tomcat is an open source software implementation of the Java Servlet specifications are developed under the [Java Community Process](#).' It also mentions that Apache Tomcat is developed in an open and participatory environment and provides a link to 'click here'. Below this, it states that Apache Tomcat powers numerous large-scale, mission-critical web applications and lists their stories on the 'PoweredBy' wiki page. A section titled 'Tomcat 6.0.41 Released' announces the release of version 6.0.41, listing notable changes: 'Add support for using ecj-P20140317-1600.jar to use Java 8 syntax in JS' and 'Update native library to 1.1.30'.

2.- Buscamos una versión para descargar

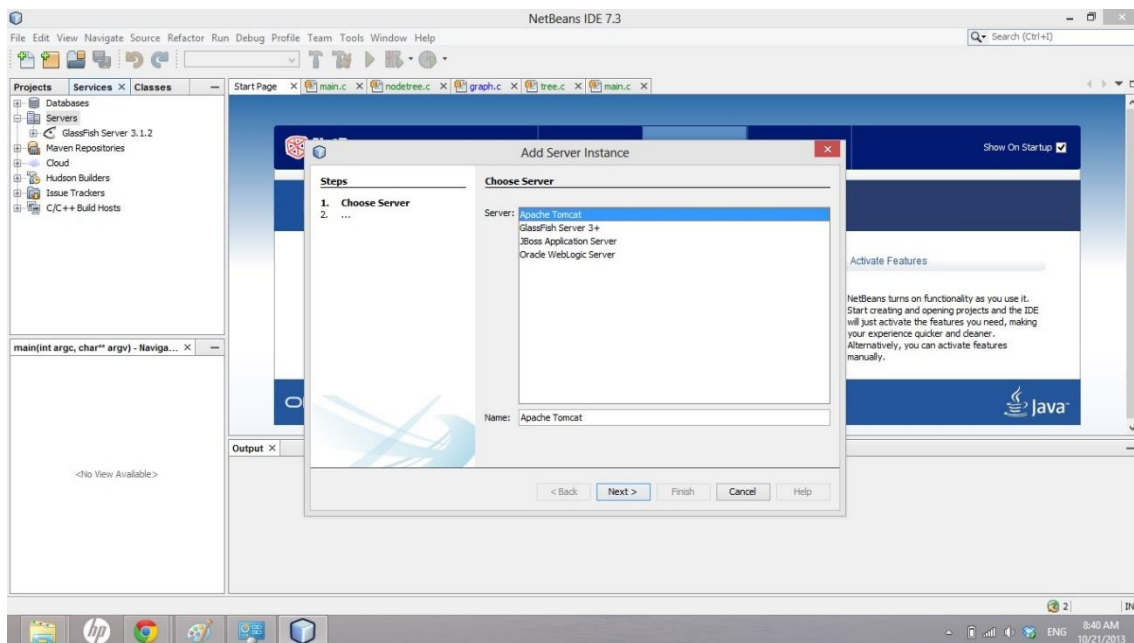


The screenshot shows the Apache Tomcat download page in a web browser. The browser's address bar displays 'tomcat.apache.org/download-70.cgi'. The page content includes a message about using a mirror: 'You are currently using <http://mirrors.ibiblio.org/apache/>. If you encounter a problem with this mirror, please select another mirror. If all mirrors are failing, there are [backup mirrors](#) (at the end of the mirrors list) that should be available.' Below this, there is a section titled '7.0.42' with a link to the 'README' file for packaging information. The 'Binary Distributions' section lists various download options: 'Core' (zip, tar.gz, 32-bit Windows zip, 64-bit Windows zip, 64-bit Itanium Windows zip, 32-bit 64-bit Windows Service Installer), 'Full documentation' (tar.gz), 'Deployer' (zip, tar.gz), 'Extras' (IMX Remote jar, Web services jar, JUnit adapters jar, JUnit log4j jar), and 'Embedded' (tar.gz, zip). The 'Source Code Distributions' section lists 'tar.gz' and 'zip' options. The browser's taskbar at the bottom shows the system clock as 8:58 AM on 10/21/2013.

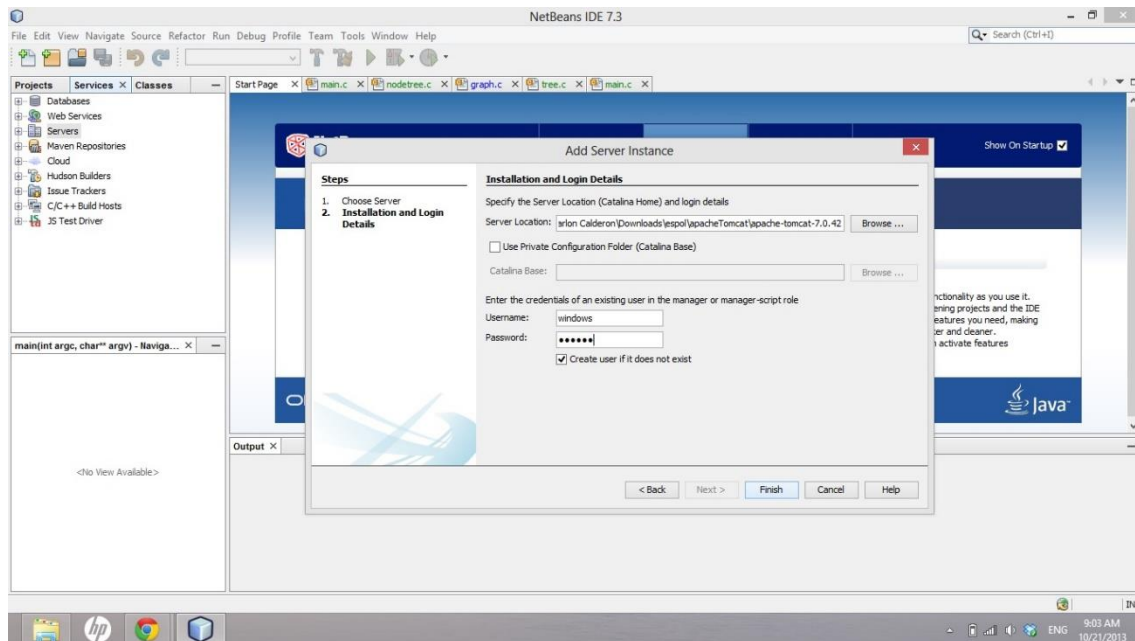
3.- Descargamos tanto el zip como el instalador



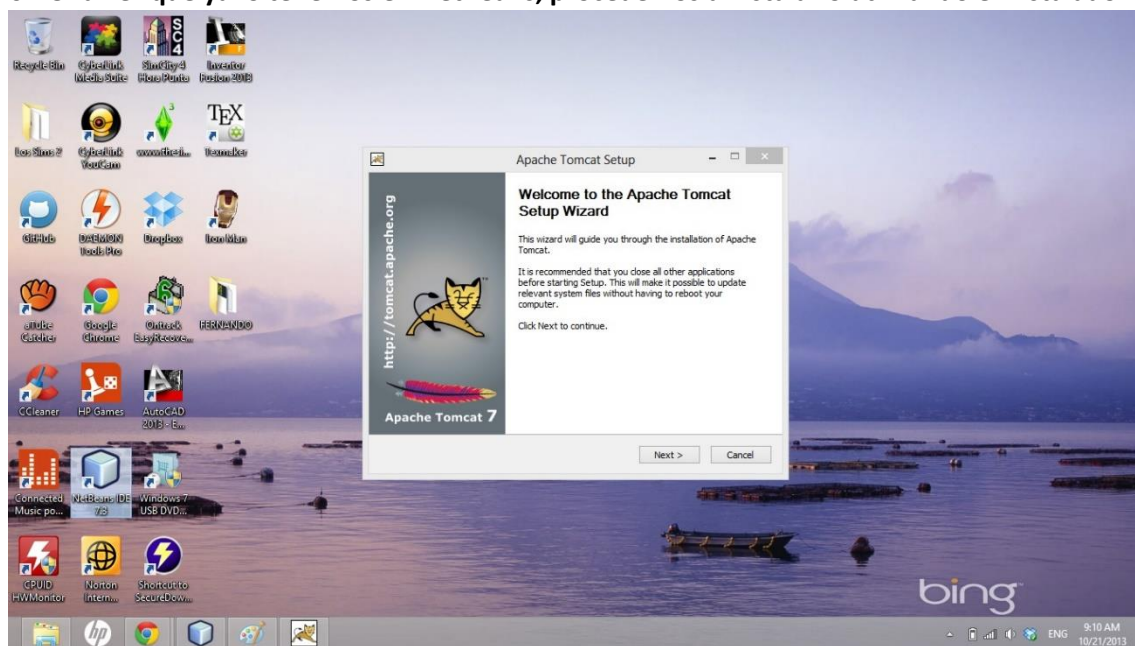
4.- Abrimos NetBeans y nos vamos a la sección services, en servers, escogemos agregar server (add server) escogemos Apache Tomcat



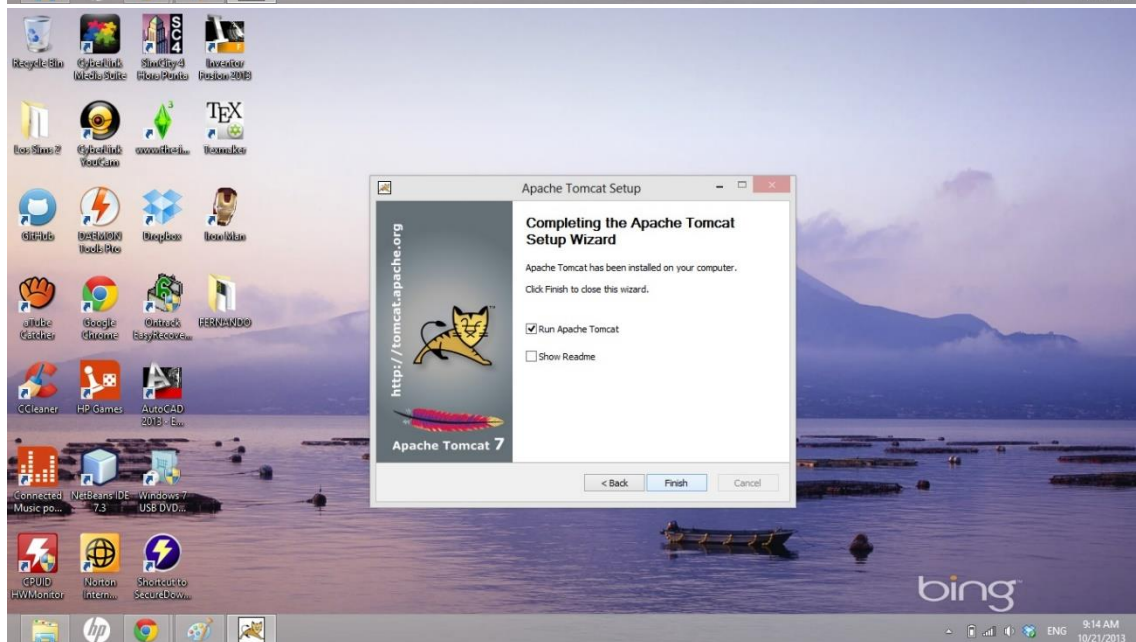
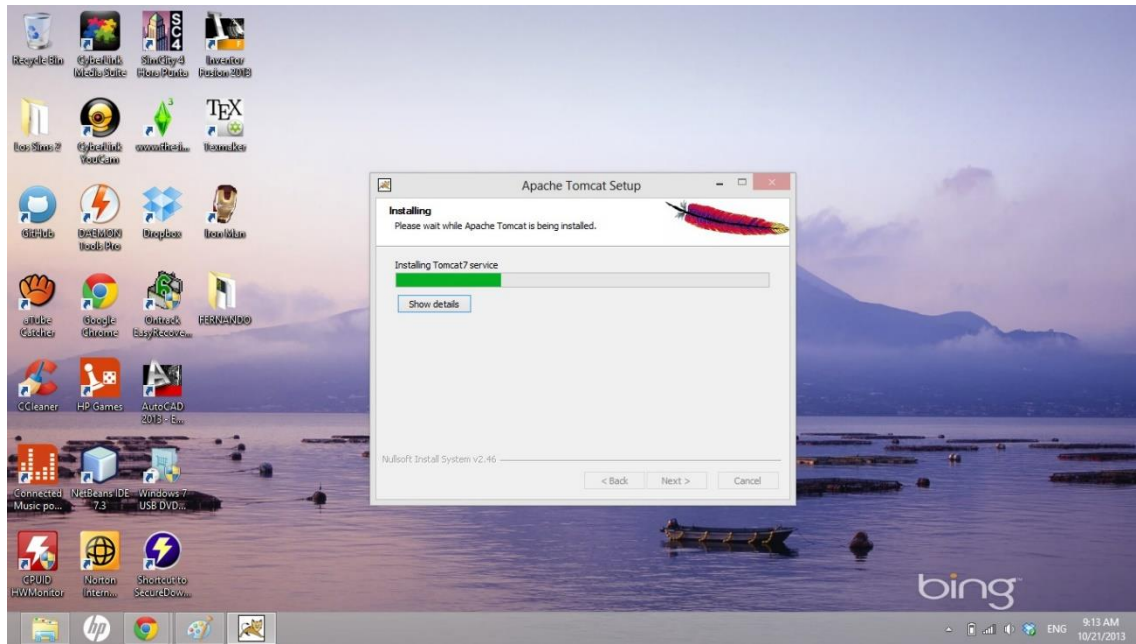
5.-Buscamos el directorio del apache tomcat y lo ingresamos, le damos un usuario y una clave.

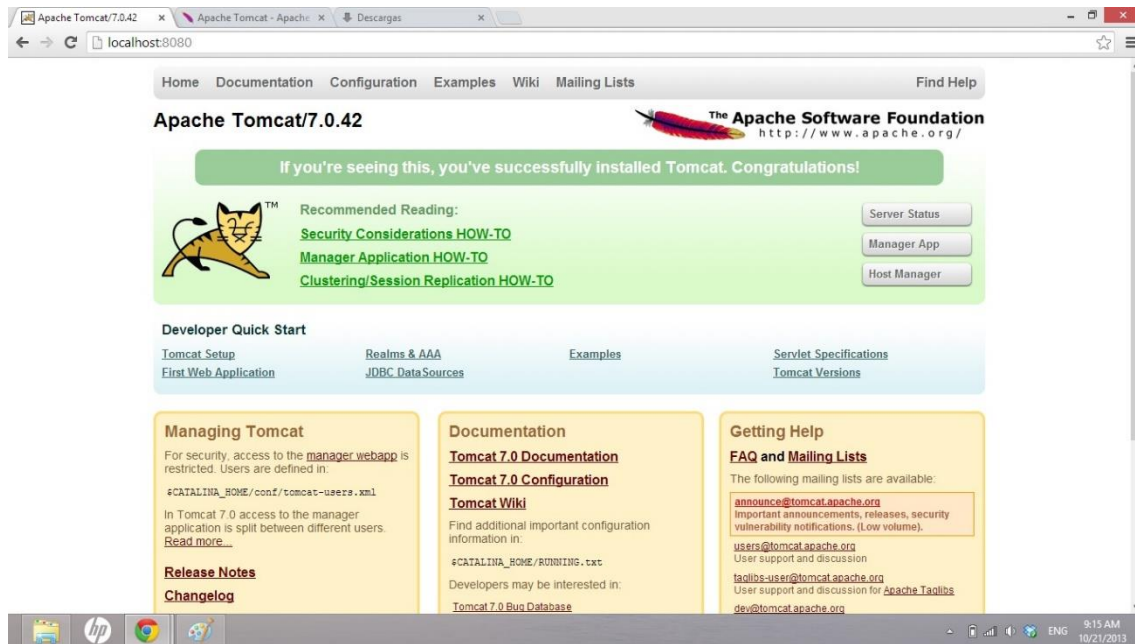


6.- Una vez que ya lo tenemos en NetBeans, procedemos a instalarlo utilizando el instalador.

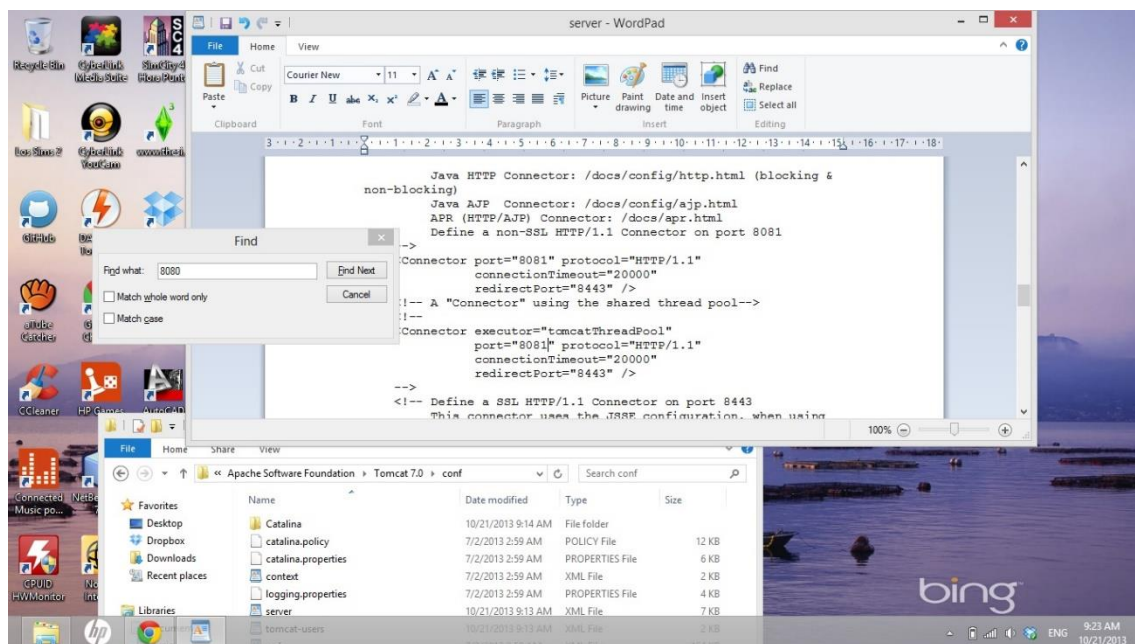


7.- Si Seguimos la instalación por defecto tendremos lo siguiente





8.- Procedemos a cambiar el puerto de nuestro apache tomcat, ya que estamos utilizando el 8081. Para esto basta con cambiar una línea en el documento server.xml



9.- Reiniciamos apache y lo volvemos a ver en el navegador, solo que usando el puerto 8081


Apache Tomcat/7.0.42

localhost:8081


HomeDocumentationConfigurationExamplesWikiMailing Lists

Find Help

Apache Tomcat/7.0.42

The Apache Software Foundation
http://www.apache.org/

If you're seeing this, you've successfully installed Tomcat. Congratulations!



Recommended Reading:

[Security Considerations HOW-TO](#)

[Manager Application HOW-TO](#)

[Clustering/Session Replication HOW-TO](#)

Server Status

Manager App

Host Manager

Developer Quick Start

[Tomcat Setup](#)

[First Web Application](#)

[Realms & AAA](#)

[JDBC Data Sources](#)

[Examples](#)

[Servlet Specifications](#)

[Tomcat Versions](#)

Managing Tomcat

For security, access to the `manager.webapp` is restricted. Users are defined in:
`$CATALINA_HOME/conf/tomcat-users.xml`
In Tomcat 7.0 access to the manager application is split between different users.
[Read more](#)

[Release Notes](#)

[Changelog](#)

Documentation

[Tomcat 7.0 Documentation](#)

[Tomcat 7.0 Configuration](#)

[Tomcat Wiki](#)

Find additional important configuration information in:
`$CATALINA_HOME/RUNNING.txt`
Developers may be interested in:
[Tomcat 7.0 Bug Database](#)

Getting Help

[FAQ and Mailing Lists](#)

The following mailing lists are available:
announce@tomcat.apache.org
Important announcements, releases, security vulnerability notifications. (Low volume).
users@tomcat.apache.org
User support and discussion
taglibs-user@tomcat.apache.org
User support and discussion for [Apache Taglibs](#)
dev@tomcat.apache.org



ENG

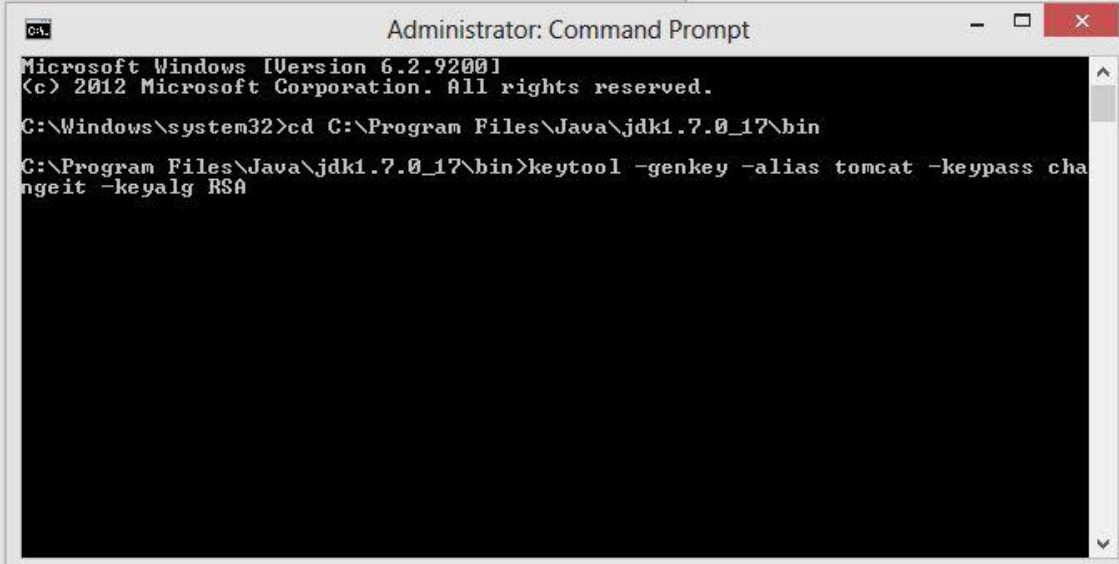
9:39 AM

10/21/2013

Creación del certificado

Creando certificado

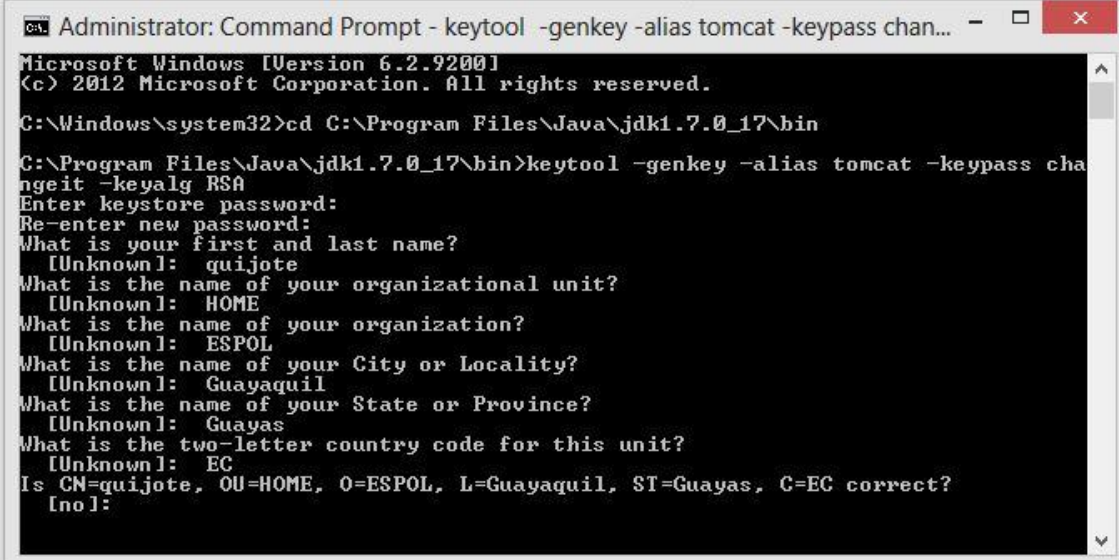
1. Trabajamos directamente con el CMD, es necesario abrirlo en modo de administrador. Ingresamos el comando para crear, el cual se ve en la captura de la imagen, es de gran importancia aclarar que se trabaja en la ruta donde se encuentra el jdk previamente instalado.



```
Administrator: Command Prompt
Microsoft Windows [Version 6.2.9200]
(c) 2012 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd C:\Program Files\Java\jdk1.7.0_17\bin
C:\Program Files\Java\jdk1.7.0_17\bin>keytool -genkey -alias tomcat -keypass cha
ngeit -keyalg RSA
```

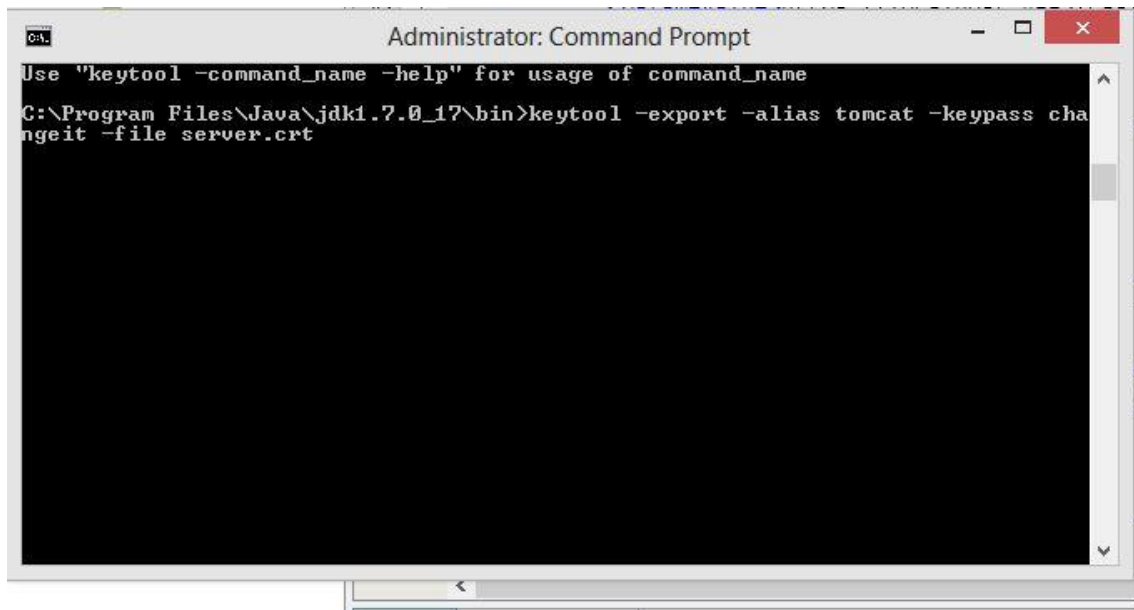
2.- Llenamos los datos que se nos van solicitando



```
Administrator: Command Prompt - keytool -genkey -alias tomcat -keypass chan...
Microsoft Windows [Version 6.2.9200]
(c) 2012 Microsoft Corporation. All rights reserved.

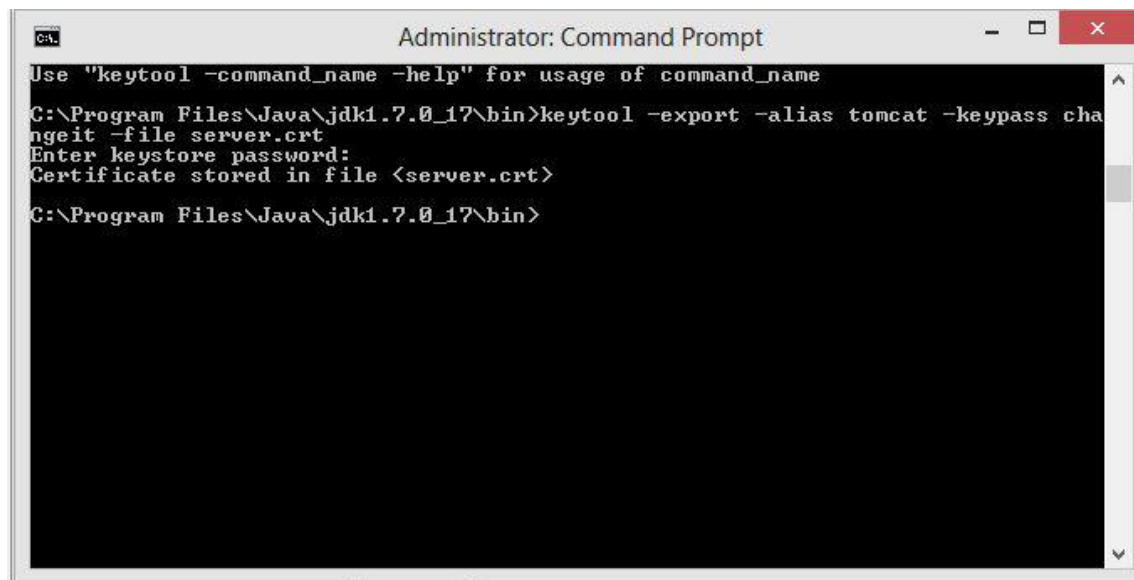
C:\Windows\system32>cd C:\Program Files\Java\jdk1.7.0_17\bin
C:\Program Files\Java\jdk1.7.0_17\bin>keytool -genkey -alias tomcat -keypass cha
ngeit -keyalg RSA
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: quijote
What is the name of your organizational unit?
[Unknown]: HOME
What is the name of your organization?
[Unknown]: ESPOL
What is the name of your City or Locality?
[Unknown]: Guayaquil
What is the name of your State or Province?
[Unknown]: Guayas
What is the two-letter country code for this unit?
[Unknown]: EC
Is CN=quijote, OU=HOME, O=ESPOL, L=Guayaquil, ST=Guayas, C=EC correct?
[no]:
```

3.- Luego usamos el comando que se ve en la imagen, comando para exportar el certificado



```
Administrator: Command Prompt
Use "keytool -command_name -help" for usage of command_name
C:\Program Files\Java\jdk1.7.0_17\bin>keytool -export -alias tomcat -keypass changeit -file server.crt
```

4.- El certificado creado, será guardado



```
Administrator: Command Prompt
Use "keytool -command_name -help" for usage of command_name
C:\Program Files\Java\jdk1.7.0_17\bin>keytool -export -alias tomcat -keypass changeit -file server.crt
Enter keystore password:
Certificate stored in file <server.crt>
C:\Program Files\Java\jdk1.7.0_17\bin>
```

5.- Ingresamos el comando que se ve en la imagen, comando para importar el certificado

```
Administrator: Command Prompt

Use "keytool -command_name -help" for usage of command_name

C:\Program Files\Java\jdk1.7.0_17\bin>keytool -export -alias tomcat -keystore cha
ngeit -file server.crt
Enter keystore password:
Certificate stored in file <server.crt>

C:\Program Files\Java\jdk1.7.0_17\bin>keytool -import -file server.crt -keypass
changeit -keystore ..\jre\lib\security\cacerts
```

6.- Indicamos que el certificado es de confianza y termina la creación del certificado

```
Administrator: Command Prompt

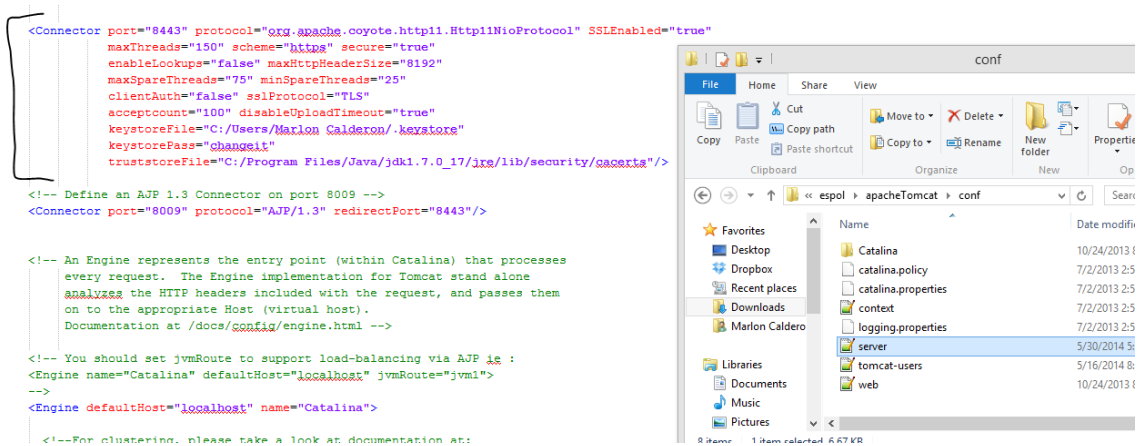
Enter keystore password:
Re-enter new password:
Owner: CN=quijote, OU=HOME, O=ESPOL, L=Guayaquil, ST=Guayas, C=EC
Issuer: CN=quijote, OU=HOME, O=ESPOL, L=Guayaquil, ST=Guayas, C=EC
Serial number: 416c72dc
Valid from: Fri May 30 16:22:13 COT 2014 until: Thu Aug 28 16:22:13 COT 2014
Certificate fingerprints:
    MD5: 55:EC:A3:BD:80:91:4C:2F:1F:E3:55:63:C9:7F:A8:5C
    SHA1: 92:2E:22:BB:13:58:CD:12:6D:EF:26:AF:0C:B5:18:51:B8:74:4B:9C
    SHA256: 17:83:5D:0A:A9:37:60:AD:8A:A2:25:34:47:0B:8E:FE:86:BC:79:C8:3
CA:2C:0B:26:22:30:0C:F8:20:5C:5B
Signature algorithm name: SHA256withRSA
Version: 3

Extensions:
#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 1F 0E 3A 01 8B B1 13 A6    56 45 70 4F 95 35 ED 19    ...:.....UEp0.5..
0010: 95 13 B5 BF                                     ....
]
]

Trust this certificate? [no]: yes
Certificate was added to keystore

C:\Program Files\Java\jdk1.7.0_17\bin>
```

Para finalizar la instalación del tomcat que usaremos con el certificado y así usar el puerto 8443 como https, solamente modificamos las líneas que se ven en la imagen, las cuales son las del documento server.xml que es parte de los archivos de configuración del tomcat.



Instalación de PostgreSQL

Instalación PostgreSQL

1.- En la siguiente dirección descargamos la versión de postgresql que instalaremos, se nos dan opciones a escoger por el sistema operativo, para este caso la descarga será para Windows.

<http://www.enterprisedb.com/products-services-training/pgdownload#windows>

EnterpriseDB
ENTERPRISEDB

Login / Register | Downloads | Blog | Forums | Store | [Contact Us](#)

+1 781-357-3390 | Search our site

Products | Services | Training | Solutions | Success Stories | Resources | Partner Programs | About Us

You are here: [Home](#) / [Products](#) / [PostgreSQL Overview](#) / Download PostgreSQL

Download PostgreSQL

Please Note: Cookies should be enabled for the download process to function correctly

Installer version Version 9.3.4

Linux x86-32 | Linux x86-64 | Win x86-32 | Win x86-64 | Mac OS X

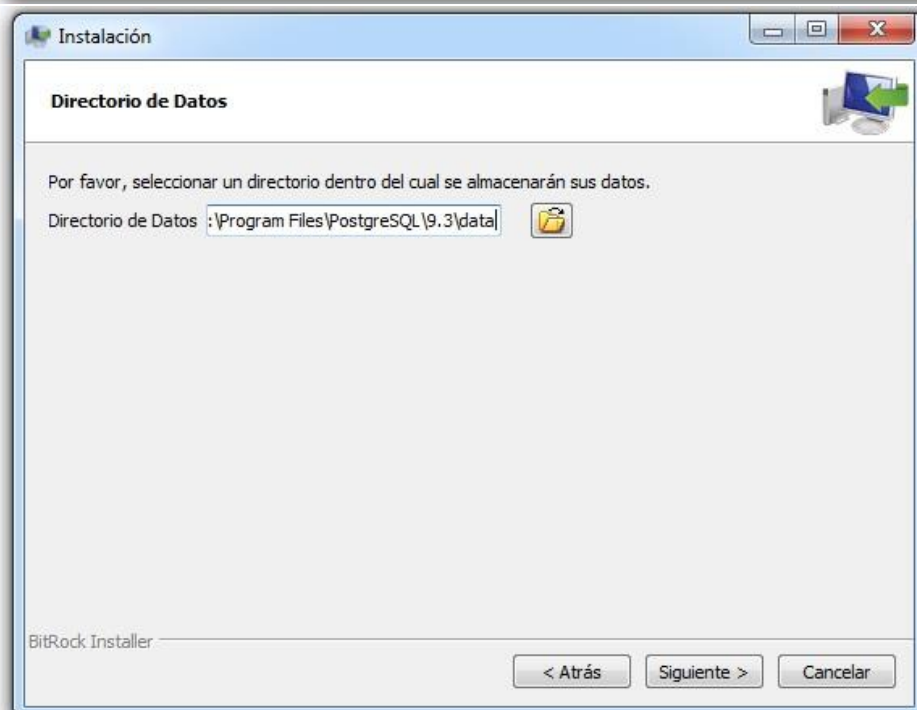
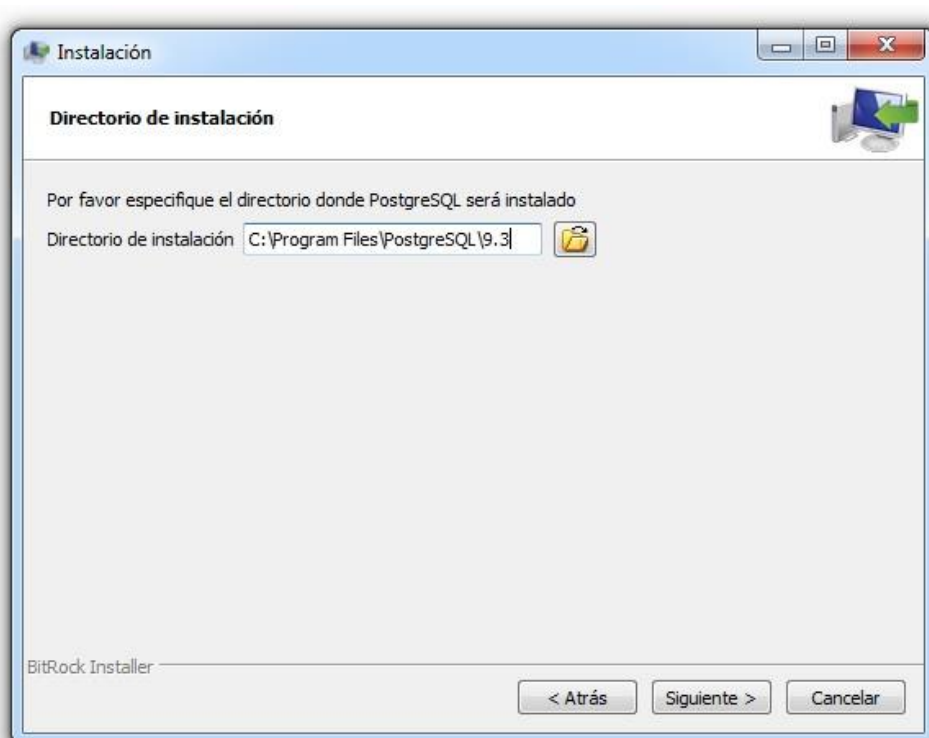
Installer version Version 9.2.8

Linux x86-32 | Linux x86-64 | Win x86-32 | Win x86-64 | Mac OS X

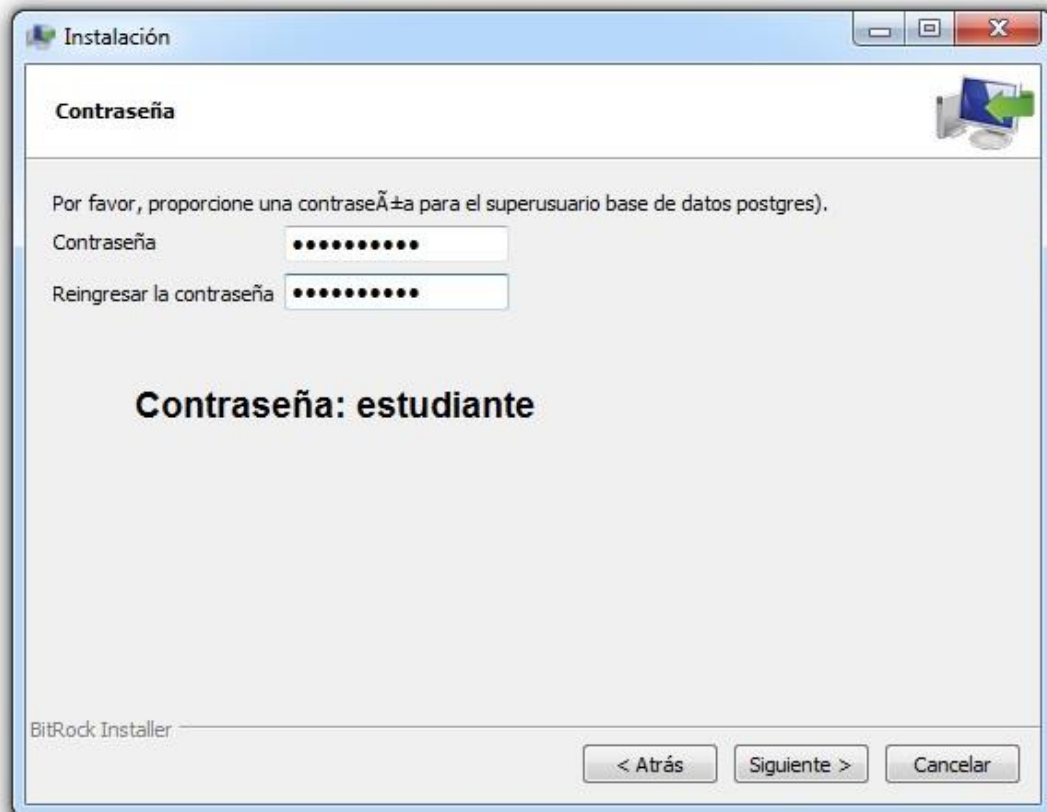
2.- Una vez descargado empezamos la instalación



3.- Se selecciona siguiente, lo hicimos con las configuraciones por defecto, hasta llegar al punto donde nos pide la contraseña.



4.-Ponemos una contraseña, nosotros le pusimos de contraseña estudiante y así mismo el resto de configuraciones se dejaron por defecto, hasta que concluyó la instalación



The screenshot shows a Windows-style installer window titled 'Instalación'. The main heading is 'Contraseña'. Below it, a message reads: 'Por favor, proporcione una contraseña para el superusuario base de datos postgres).' There are two password input fields: 'Contraseña' and 'Reingresar la contraseña', both filled with dots. Below the fields, the text 'Contraseña: estudiante' is displayed. At the bottom, the text 'BitRock Installer' is visible on the left, and three buttons '< Atrás', 'Siguiete >', and 'Cancelar' are on the right.

Instalación

Contraseña

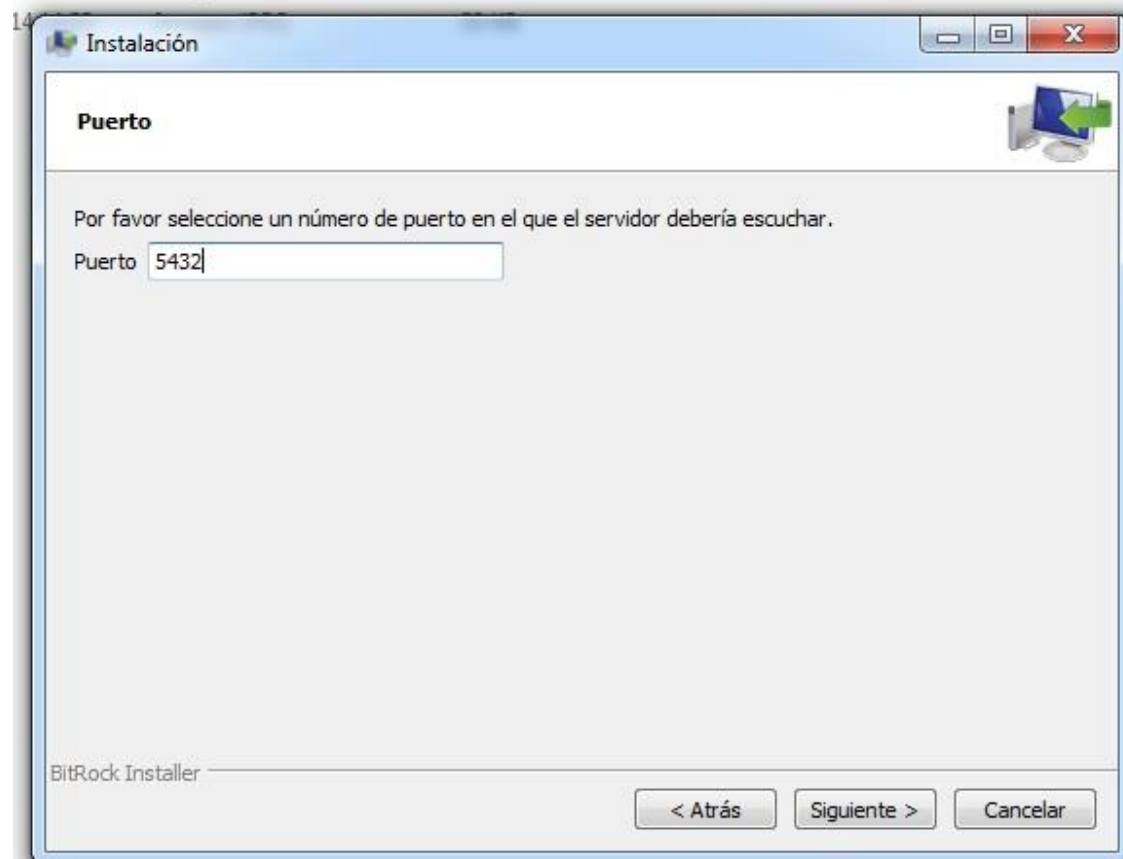
Por favor, proporcione una contraseña para el superusuario base de datos postgres).

Contraseña:
Reingresar la contraseña:

Contraseña: estudiante

BitRock Installer

< Atrás Siguiete > Cancelar



The screenshot shows a second window from the BitRock Installer, titled 'Instalación'. The main heading is 'Puerto'. Below it, a message reads: 'Por favor seleccione un número de puerto en el que el servidor debería escuchar.' There is a single input field labeled 'Puerto' containing the value '5432'. At the bottom, the text 'BitRock Installer' is visible on the left, and three buttons '< Atrás', 'Siguiete >', and 'Cancelar' are on the right.

Instalación

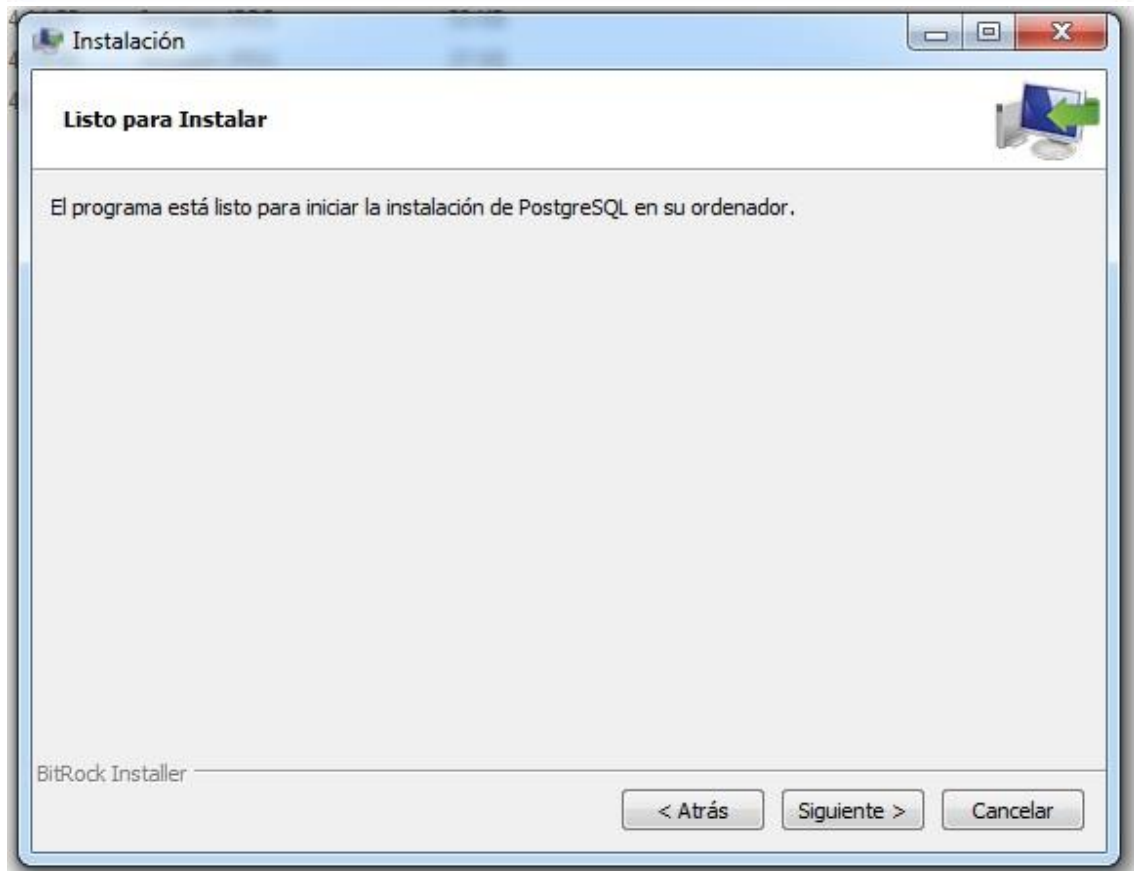
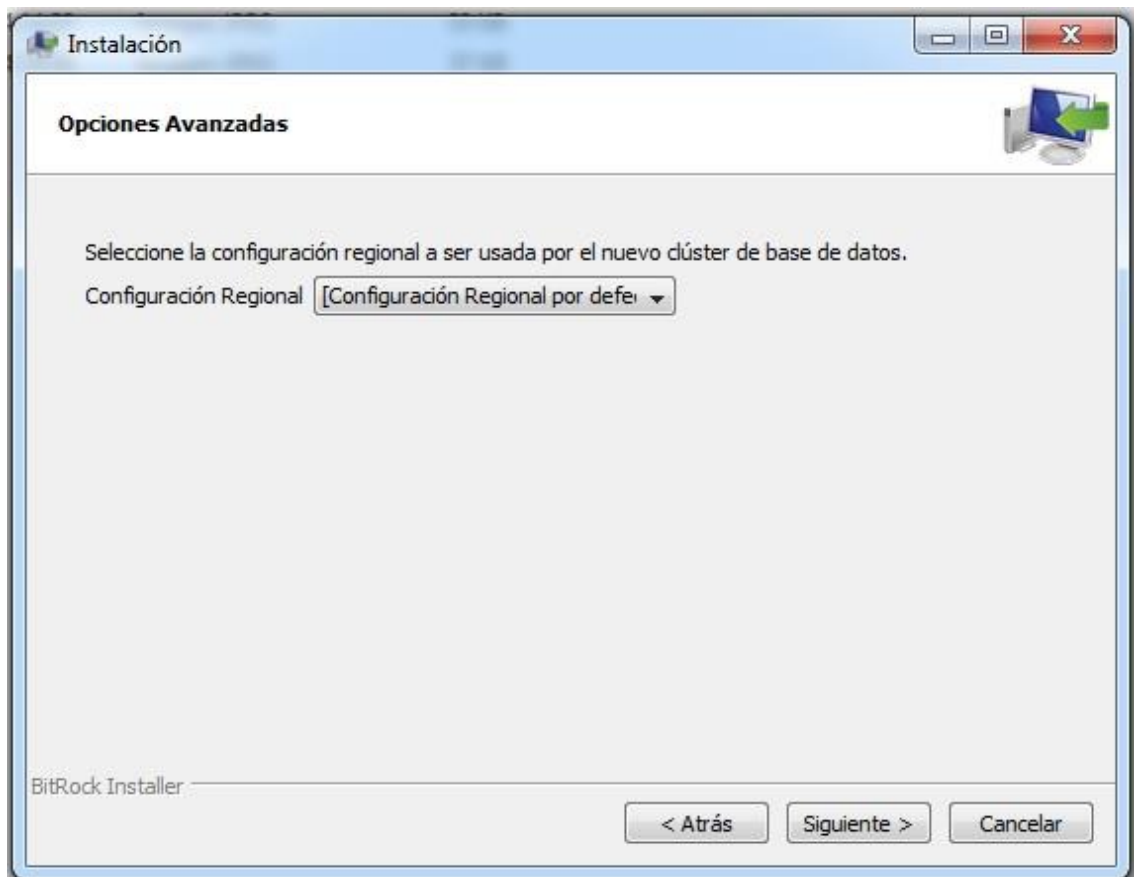
Puerto

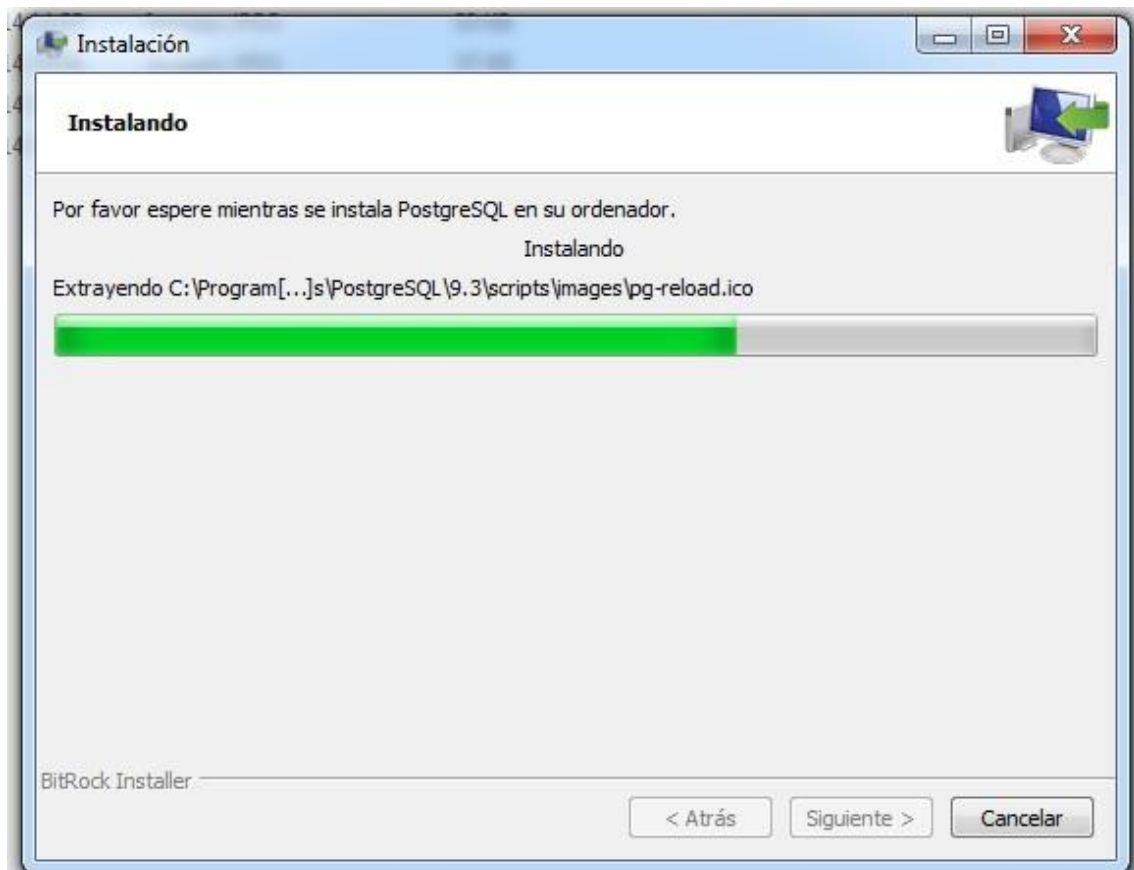
Por favor seleccione un número de puerto en el que el servidor debería escuchar.

Puerto: 5432

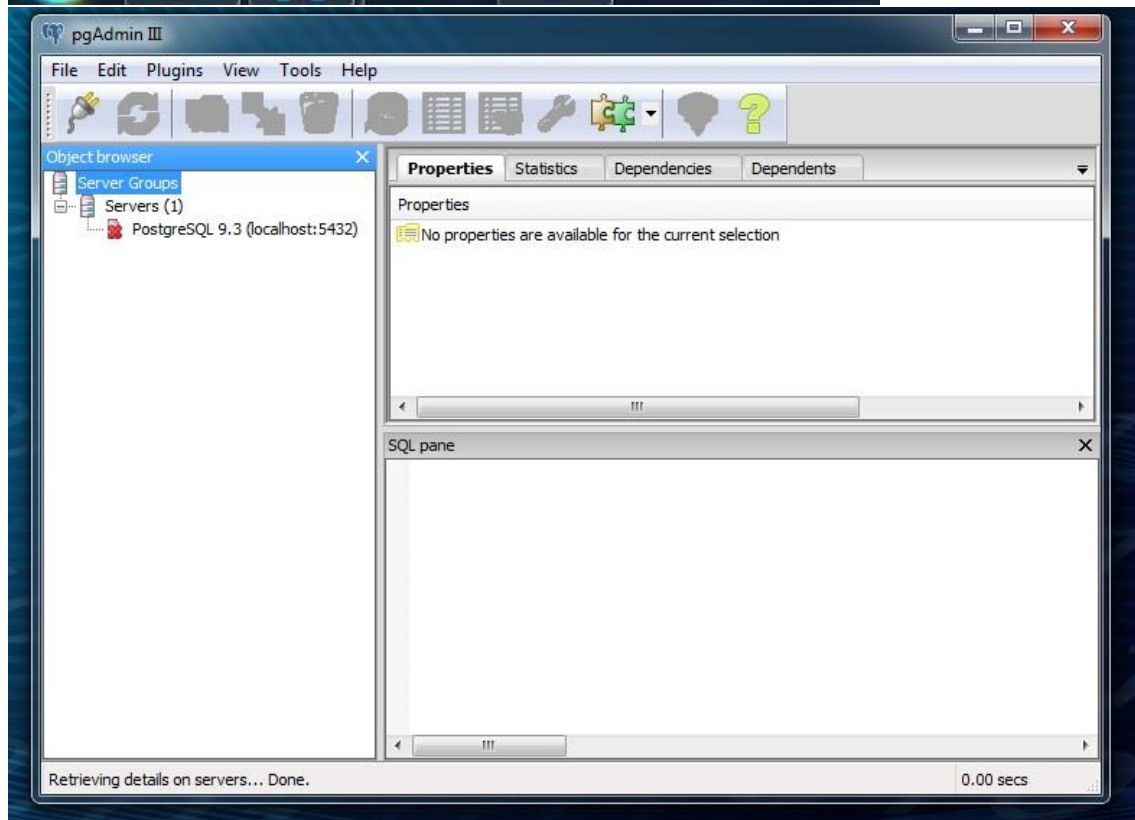
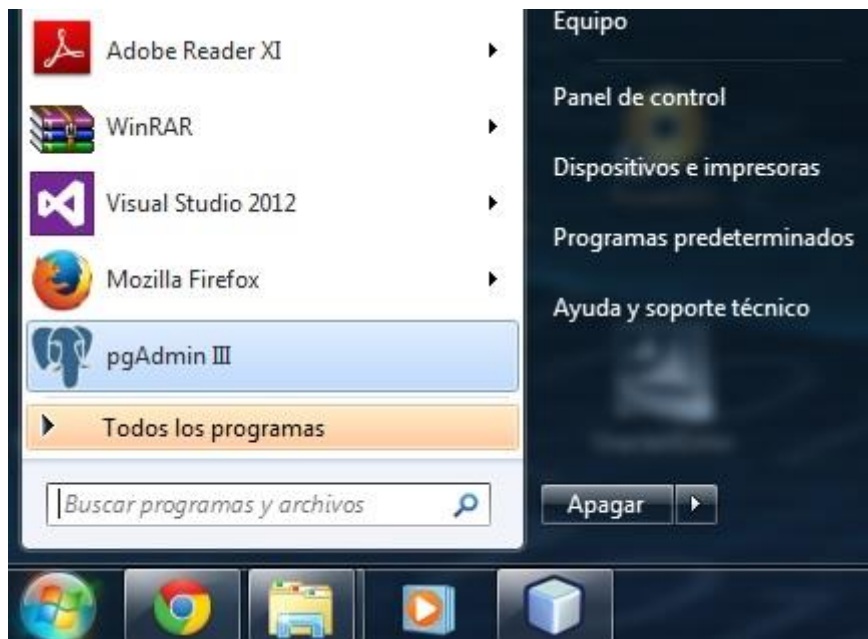
BitRock Installer

< Atrás Siguiete > Cancelar





5.- Finalmente ejecutamos el pgAdmin III para comprobar que todo esté bien instalado



Es necesaria la creación de una base de datos para la parte de la autenticación CAS

Se lo hará con el postgresQL, utilizando el siguiente script

```
-- DROP DATABASE "Quijote";

CREATE DATABASE "Quijote"
  WITH OWNER = postgres
       ENCODING = 'UTF8'
       TABLESPACE = pg_default
       LC_COLLATE = 'English_United States.1252'
       LC_CTYPE = 'English_United States.1252'
       CONNECTION LIMIT = -1;
```

Luego será necesario tener una tabla que guarde los registros de los usuarios.

```
-- Table: "Users"








-- DROP TABLE "Users";

CREATE TABLE "Users"
(
  id_user serial NOT NULL,
  username character varying(100) NOT NULL,
  password character varying(100) NOT NULL,
  nombre character varying(100) NOT NULL,
  apellido character varying(100) NOT NULL,
  CONSTRAINT id_user PRIMARY KEY (id_user)
)
WITH (
  OIDS=FALSE
);
ALTER TABLE "Users"
  OWNER TO postgres;
```

Finalmente se hace el ingreso de los datos, de la siguiente manera

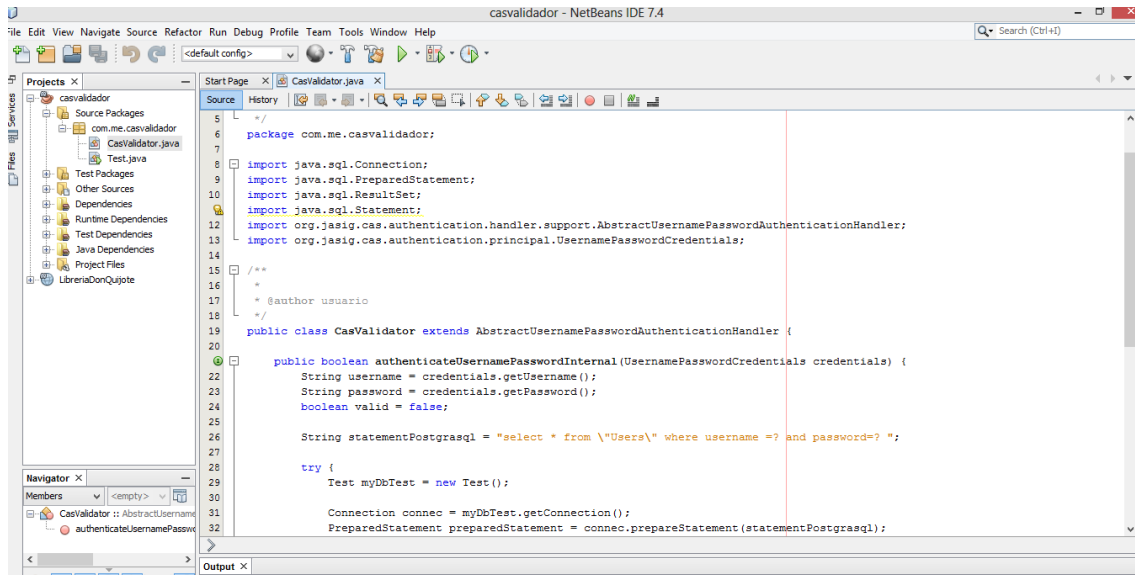
```
-- Table: "Users"

insert into "Users" (username, password, nombre, apellido)
values ('jlmj', '1234', 'Jose', 'Monar');
```

Edit Data - PostgreSQL 9.3 (localhost:5432) - Quijote - Users						
File Edit View Tools Help						
       100 rows						
	id_user [PK] integer	username character vai	password character vai	nombre character vai	apellido character vai	
1	1	admin	1234	marlon	calderon	
2	2	jlmj	1234	Jose	Monar	
*						

Integración CAS-Spring-proyecto-Base PostgreSQL

1.- Se necesita crear la clase casValidador, creando un proyecto de java bajo ese nombre.



2.- Se recomienda trabajar con los statements.

```
String statementPostgrasql = "select * from \"Users\" where username =? and password=? ";

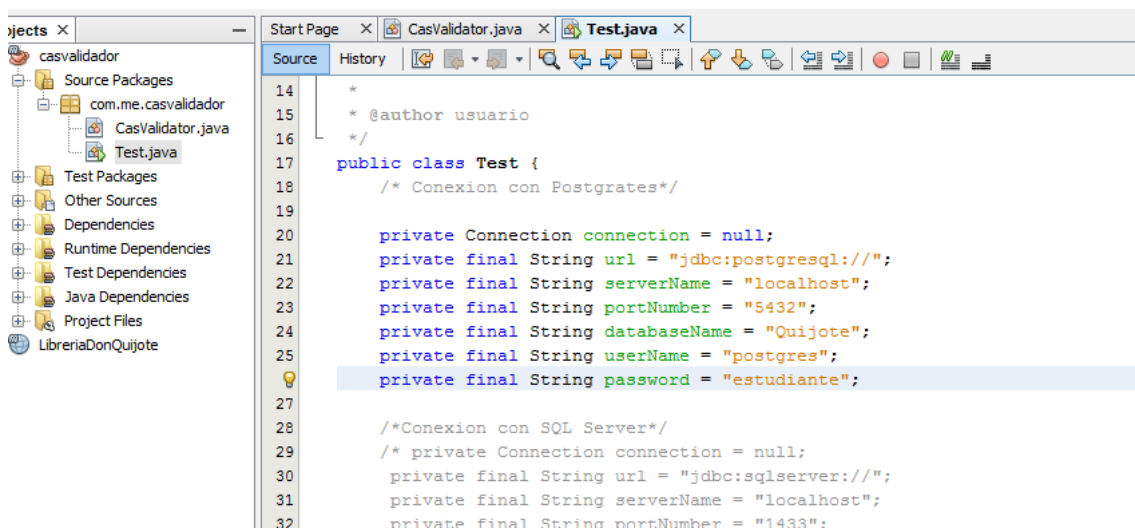
try {
    Test myDbTest = new Test();

    Connection connec = myDbTest.getConnection();
    PreparedStatement preparedStatement = connec.prepareStatement(statementPostgrasql);
    preparedStatement.setString(1, username);
    preparedStatement.setString(2, password);

    ResultSet result = preparedStatement.executeQuery();

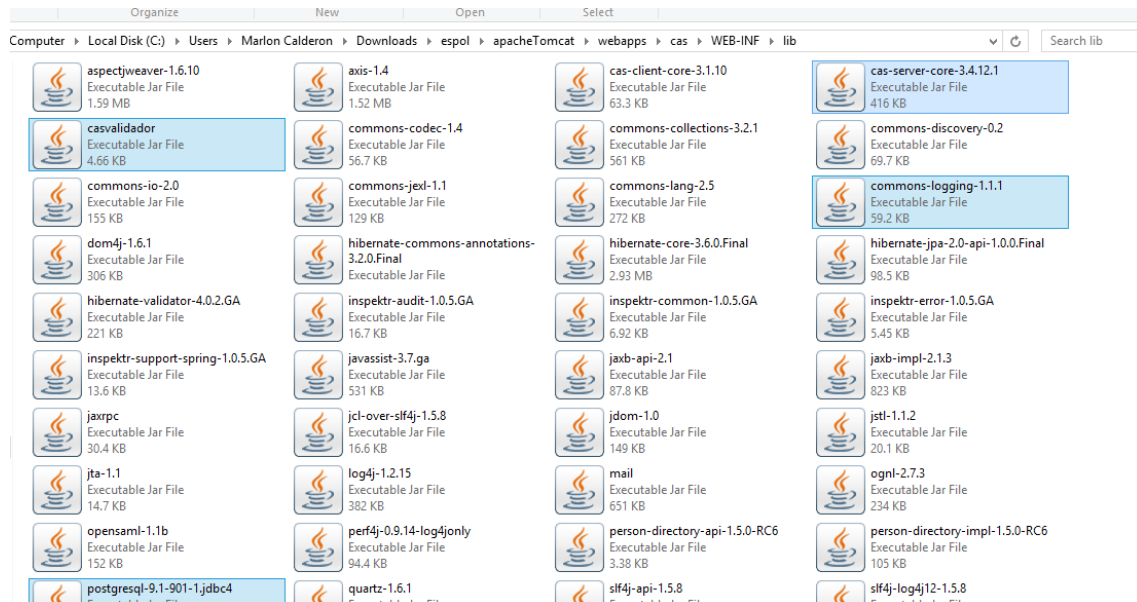
    while (result.next()) {
        valid = true;
    }
}
```

3.- Se usa una clase de main, para realizar la conexión.



4.- Este proyecto se compila y se genera de este modo un documento casvalidador.jar
Todas estas cosas están agregadas en el repositorio

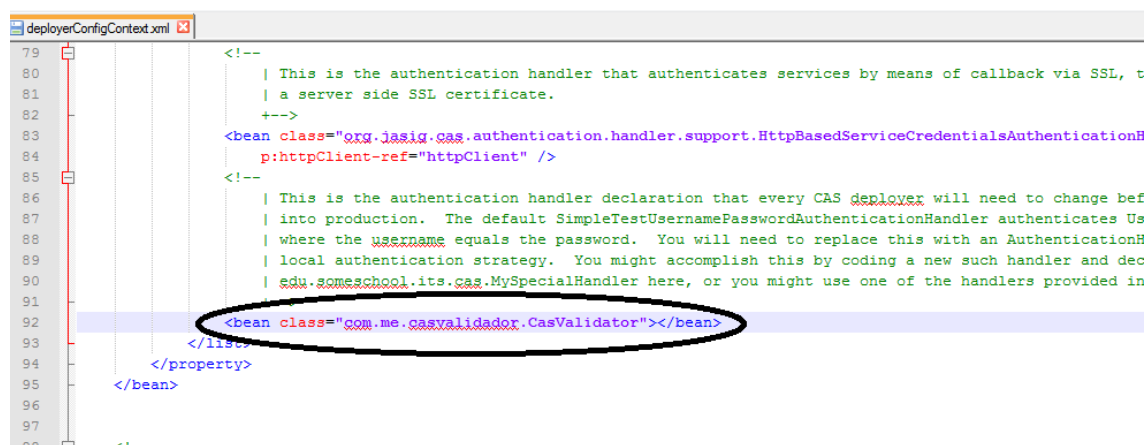
5.- Finalmente este documento se copia junto con otros que son necesarios para CAS, estos son los siguientes: cas-server-core-3.4.12.1, casvalidador.jar, commons-loggind-1.1.1, postgresql-9.1.901-1.jdbc4



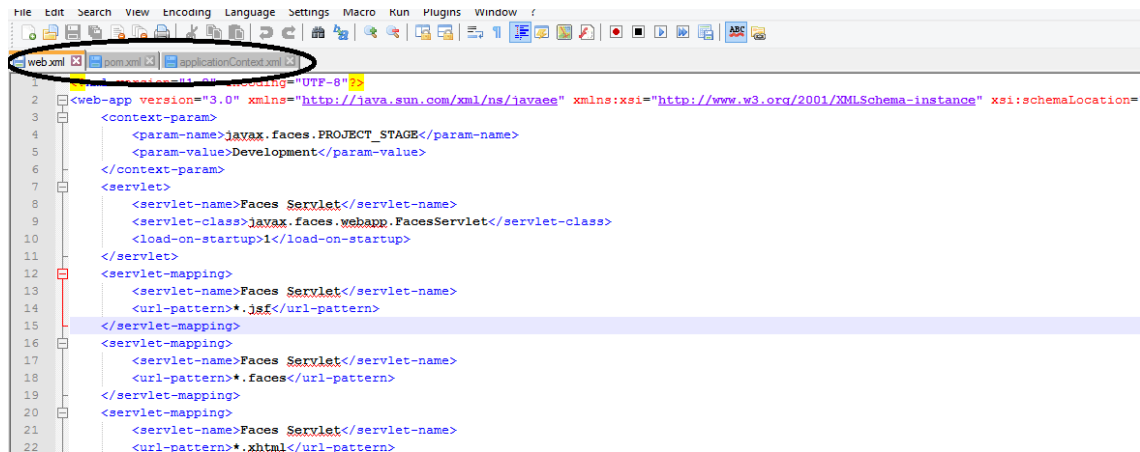
Esto va dentro de la carpeta de cas que está entre las webapps de Apache Tomcat, dentro de la parte de las librerías.

6.- Se debe cambiar el archivo deployerConfigContext.xml que se encuentra en la carpeta WEB-INF de la carpeta cas.

Se indica que se utilizará el casvalidador que hemos hecho en el paso anterior.



7.- Finalmente se realizará la configuración para el proyecto. Para esto se configuran tres archivos importantes del proyecto que son: web.xml, Pom.xml, applicationContext.xml.



En web.xml

Se agregan las siguientes líneas.

```

<context-param>
  <param-name>contextConfigLocation</param-name>
  <param-value>/WEB-INF/applicationContext.xml</param-value>
</context-param>
<listener>
  <listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>
</listener>

<filter>
  <filter-name>springSecurityFilterChain</filter-name>
  <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>
</filter>

<filter-mapping>
  <filter-name>springSecurityFilterChain</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>

```

En pom.xml

Se incluyen las siguientes dependencias.

```
<dependency>
  <groupId>commons-logging</groupId>
  <artifactId>commons-logging</artifactId>
  <version>1.1.1</version>
</dependency>
<dependency>
  <groupId>cas</groupId>
  <artifactId>casclient</artifactId>
  <version>2.1.1</version>
</dependency>
<dependency>
  <groupId>javax.servlet</groupId>
  <artifactId>jstl</artifactId>
  <version>1.2</version>
</dependency>
<dependency>
  <groupId>org.jasig.cas.client</groupId>
  <artifactId>cas-client-core</artifactId>
  <version>3.3.1</version>
</dependency>

<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-core</artifactId>
  <version>3.0.0.RELEASE</version>
  <optional>false</optional>
</dependency>
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-webmvc</artifactId>
  <version>3.0.0.RELEASE</version>
  <optional>false</optional>
</dependency>
```

```

<dependency>
  <groupId>org.springframework.security</groupId>
  <artifactId>spring-security-core</artifactId>
  <version>3.0.0.RELEASE</version>
  <optional>>false</optional>
</dependency>
<dependency>
  <groupId>org.springframework.security</groupId>
  <artifactId>spring-security-config</artifactId>
  <version>3.0.0.RELEASE</version>
  <scope>compile</scope>
</dependency>
<dependency>
  <groupId>org.springframework.security</groupId>
  <artifactId>spring-security-cas-client</artifactId>
  <version>3.0.0.RELEASE</version>
  <optional>>false</optional>
</dependency>

```

En applicationContext.xml

Se especifican las direcciones de las vistas, además de los roles, tipos de usuarios que tiene la aplicación y la configuración general de Spring.

```

<sec:http entry-point-ref="casAuthenticationEntryPoint" auto-config="true" path-type="ant">

  <sec:custom-filter before="CAS_FILTER" ref="casSingleSignOutFilter"/>
  <sec:custom-filter after="CAS_FILTER" ref="casAuthenticationFilter"/>
  <sec:intercept-url pattern="/administrador/*" access="ROLE_ADMIN"/>
  <sec:intercept-url pattern="/*" access="ROLE_VENDEDOR"/>

  <sec:logout logout-success-url="https://guijote:8443/cas/logout" invalidate-session="true"/>
</sec:http>

<!--
  Required for the casProcessingFilter, so define it explicitly set and
  specify an Id Even though the authenticationManager is created by
  default when namespace based config is used.
-->
<sec:authentication-manager alias="authenticationManager">
  <sec:authentication-provider ref="casAuthenticationProvider" />
</sec:authentication-manager>

```

```

-->
<bean id="serviceProperties" class="org.springframework.security.cas.ServiceProperties">
  <property name="service" value="http://guijote:8084/pruebaCAS/_spring_cas_security_check"/>
  <property name="sendRenew" value="false"/>
</bean>

<!--
  The CAS filter handles the redirect from the CAS server and starts the ticket validation.
-->
<bean id="casAuthenticationFilter" class="org.springframework.security.cas.web.CasAuthenticationFilter">
  <property name="authenticationManager" ref="authenticationManager"/>
</bean>

<!--
  Handles a logout request from the CAS server
-->
<bean id="casSingleSignOutFilter" class="org.jasig.cas.client.session.SingleSignOutFilter" />

<!--
  The entryPoint intercepts all the CAS authentication requests.
  It redirects to the CAS loginUrl for the CAS login page.
-->
<bean id="casAuthenticationEntryPoint" class="org.springframework.security.cas.web.CasAuthenticationEntryPoint">
  <property name="loginUrl" value="https://guijote:8443/cas/login"/>
  <property name="serviceProperties" ref="serviceProperties"/>
</bean>

<!--
  Handles the CAS ticket processing.
-->
<bean id="casAuthenticationProvider" class="org.springframework.security.cas.authentication.CasAuthenticationProvider">
  <property name="userDetailsService" ref="userServices"/>
  <property name="serviceProperties" ref="serviceProperties"/>
  <property name="ticketValidator">
    <bean class="org.jasig.cas.client.validation.Cas20ServiceTicketValidator">
      <constructor-arg index="0" value="https://guijote:8443/cas"/>
    </bean>
  </property>
  <property name="key" value="cas"/>
</bean>

<!--
  The users available for this application.
-->
<sec:user-service id="userServices">
  <sec:user name="admin" password="1234" authorities="ROLE_ADMIN, ROLE_VENDEDOR"/>
  <sec:user name="jimi" password="1234" authorities="ROLE_VENDEDOR"/>
</sec:user-service>

```

