

**Desarrollo de una aplicación
J2EE para la gestión del
control de presencia**

Memoria



CAROLINA SANTANA GONZÁLEZ

Consultor: Antoni Oller Arcas

11/01/2013

Índice de contenido

1. Introducción.
2. Descripción, Objetivo y Planificación del TFC.
 - 2.1 Justificación y descripción del TFC.
 - 2.2 Objetivos.
 - 2.3 Planificación.
3. Análisis.
 - 3.1 Componentes.
 - 3.2 Descripción de funcionamiento.
 - 3.3 Requerimientos funcionales y casos de uso.
4. Diseño.
 - 4.1. Interfaz de usuario.
 - 4.2. Tecnología utilizada.
 - 4.3. Imágenes capturadas en funcionamiento
5. Manuales.
 - 5.1 Guía para crear el proyecto.
 - 5.2 Guía para configurar Spring e instalar, configurar y utilizar la consola Hibernate.
 - 5.2.1 Configuración spring.xml - hibernate.cfg.xml
 - 5.2.2 Instalación hibernate-tools.
 - 5.2.3 Configurar y utilizar la consola de hibernate.
6. Conclusiones.

1. Introducción.

La motivación principal en la realización de este proyecto es diseñar y programar una aplicación útil que creo que podría servir como base y prototipo para otras aplicaciones.

Se me ocurrió realizar un proyecto web para la gestión del control de presencia, ya que es una necesidad inminente en la empresa en la que trabajo. Además para mí, ha supuesto la rampa de salida para el conocimiento y la práctica en el manejo del conjunto de tecnología del que se compone.

2. Descripción, objetivos y planificación del TFC.

2.1 Justificación y descripción del TFC.

Ya son de sobra conocidos en cualquier empresa los dispositivos que indican la presencia de los trabajadores en sus dependencias, pero muchos de ellos no son capaces de computar el tiempo neto que ha durado la jornada realmente. Se limitan a presentar el momento de la entrada y la salida.

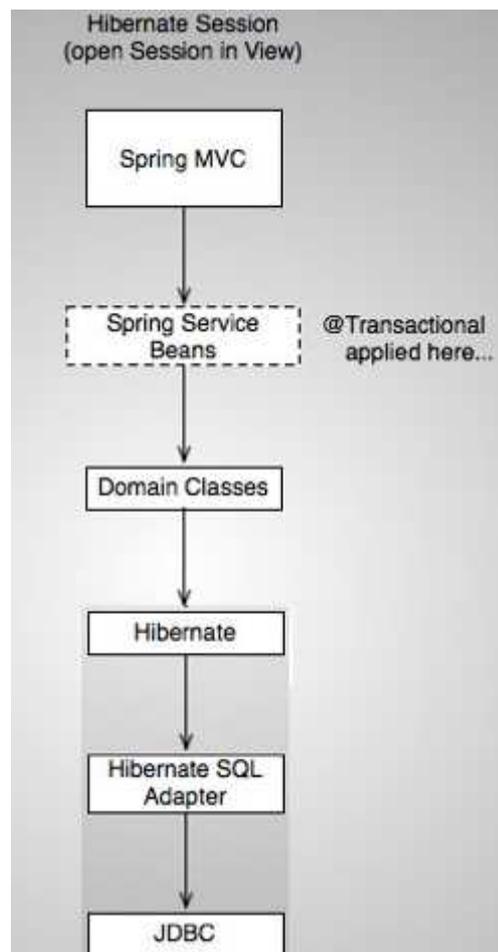
En los dispositivos más complejos que admitan cierto tipo de incidencias, éstos están sujetos a las equivocaciones que podamos cometer y a cualquier eventualidad que no pueda ser registrada.

Una buena alternativa sería tener una aplicación para revisar los errores que se puedan cometer al utilizar este tipo de dispositivos y también facilite registrar nuevas incidencias computables a efectos de jornada que el sistema no hay sido capaz de detectar.

2.2 Objetivos.

Este Trabajo de Fin de Carrera muestra el desarrollo de una aplicación J2EE. No se pretende sea solo un programa con sus fases y etapas de diseño, desarrollo y demás, sino que sirva de guía para otras aplicaciones.

Objetivos específicos.



Con los estándares de JEE para el desarrollo de aplicaciones utilizando el **patrón de diseño MVC** (modelo vista controlador) se quiere conseguir poner en práctica los conocimientos de **Hibernate con anotaciones** para el mapeo de tablas de la base de datos.

hibernatePostgreCpresen.cfg.xml
hibernatePostgreCpresen.reveng.xml

También se pretende conectar y hacer transacciones en la base de datos utilizando las posibilidades que nos brinda **Spring**.

spring.xml

Y por último, con **JSF 2.0**, comprobar el uso de plantillas y también que ya no es necesario un fichero xml de configuración para navegación entre páginas, solo dichas páginas en si.

Todas las páginas → .xhtml

2.3 Planificación.

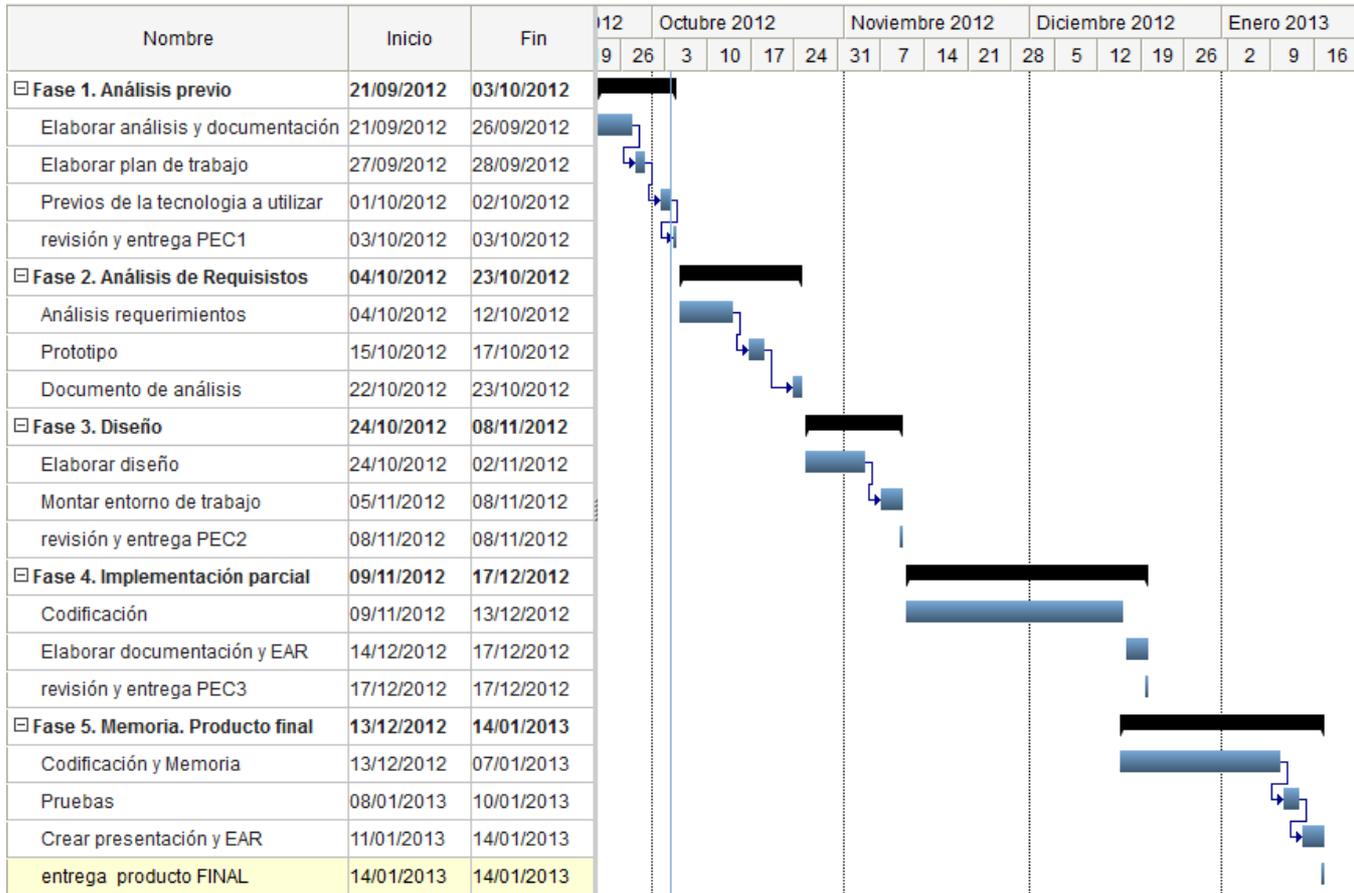
El proyecto comenzó el día 21/09/2012 y finalizará el día 14/01/2013.

Resumen de las fechas clave.

<i>Entregas</i>	<i>Documento</i>	<i>Descripción</i>
03/10/2012	PEC1	Plan de trabajo. Objetivos y planificación
08/11/2012	PEC2	Análisis funcional y de requerimientos
17/12/2012	PEC3	Implementación parcial del proyecto
14/01/2013	MEMORIA	Síntesis, presentación y producto final.

La distribución detallada de las tareas, su duración estimada, así como su plazo temporal se puede apreciar en el siguiente punto:

Distribución detallada



3. Análisis.

Este proyecto permitirá, para el personal y para los trabajadores que no siendo empleados, se incorporan eventualmente a nuestras instalaciones, gestionar el control de presencia. Para ello trataremos los siguientes procedimientos:

- Login de la aplicación
- Carga y proceso de datos
- Consultas
- Informes

3.1 Componentes del proyecto.

La aplicación está dividida en una serie de módulos, cada uno de ellos con autonomía propia.

El módulo de Login será el punto de entrada de la aplicación para cualquier **trabajador** o **administrador**.

El módulo de Carga de datos será el encargado de introducir los datos del reloj en el sistema. Accesible sólo para el personal **administrador**.

El módulo de Proceso de datos será en el que se hagan las rectificaciones pertinentes para el adecuado cómputo de horas. Accesible sólo para el personal **administrador**.

Por último, el módulo de Consultas e Informes, en el que podremos obtener información del cómputo de horas. Accesible al personal **administrador** y al **trabajador**.

3.2 Descripción de funcionamiento.

En primera instancia, cada usuario se logea en la aplicación, la cual permitirá acceder a unos módulos u otros según sea un usuario **administrador** o **trabajador**.

Un **administrador** puede hacer consulta e informes de cualquier trabajador y además tiene acceso a los módulos de carga y proceso de datos.

La primera vez que se entra en el sistema, debe ser un usuario de este tipo, para que pueda hacer la primera carga de datos para que éstos puedan ser consultados o procesados. De no ser así, simplemente el usuario trabajador no encontrará datos.

La carga de datos será por tramos de fechas. En este momento y totalmente transparente para el usuario, se hacen una serie de cálculos con los datos originales para generar la información necesaria para el procedo de datos.

Los datos quedarán en estado cargado o pendiente (si los cálculos impiden que la información se quede correcta); tal como indicamos:

1 habrá empleados que tengas todos los días con los datos correctos y no sea necesario procesar nada.

Estos datos estarían en estado cargado.

2 habrá empleados que tenga todo correcto pero que falte añadir tiempo trabajado indetectable por el sistema o computar tiempo justificable.

Estos datos estarían en estado cargado.

3 habrá empleados que tengan datos con días erróneos por causas como que el empleado se equivoca al fichar o simplemente se olvida, etc. Habría que modificar.

Estos datos estarían en estado pendiente.

El proceso de datos. En la carga de datos se genera la información necesaria para poder hacer el proceso de datos de la forma adecuada.

Es en el resultado de los puntos 2 y 3 de la carga de datos donde es necesaria la intervención de **administrador**, donde por empleado y fecha irá justificado o modificando los datos. Los datos tratados pasarán a estado cargado.

El usuario **trabajador** tendrá la posibilidad de consultar y verificar el total de horas computables aunque solo hasta la fecha en que los datos hayan sido procesados. También tiene la posibilidad de obtener un pequeño informe.

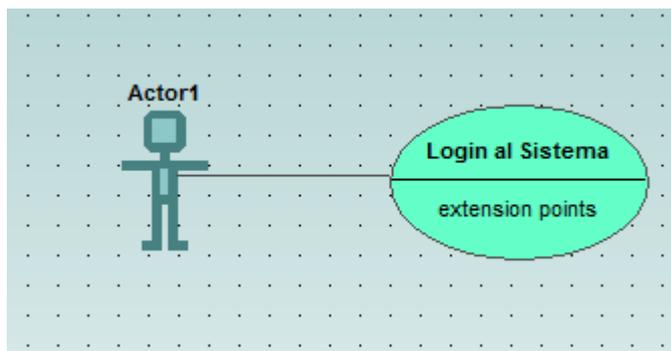
3.3 Requerimientos funcionales.

La aplicación está dividida en cuatro módulos.

3.3.1. Módulo Login.

Nos permite identificar el usuario en el sistema y darle acceso según su perfil al resto de módulos que le correspondan.

Gráfico de los casos de uso.



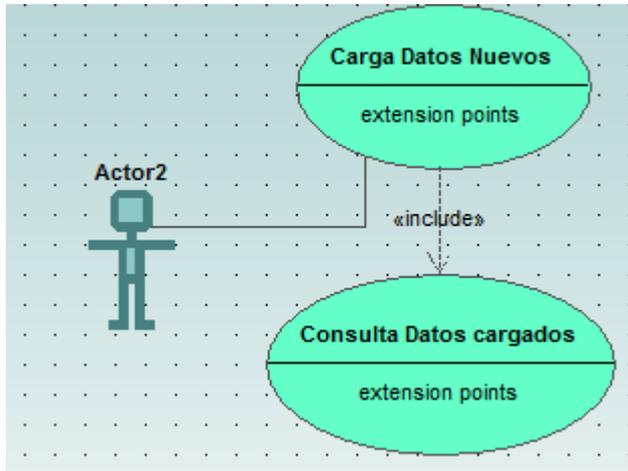
Descripción textual de los casos de uso.

CASO DE USO: "Login al Sistema"	
Funcionalidad	Punto de entrada al sistema
Actores	Cualquier usuario: administrador, trabajador
Casos de uso relacionados	
Precondición	
Flujo normal	El usuario introduce login y contraseña correctos y entra en el sistema
Flujo alternativo	Login y/o contraseña incorrectos. No entra en el sistema.
Postcondición	El usuario queda conectado al sistema

3.3.2. Módulo Carga de datos.

Accesible solo al personal autorizado, incorpora los datos originales del reloj al sistema.

Gráfico de los casos de uso.



Descripción textual de los casos de uso.

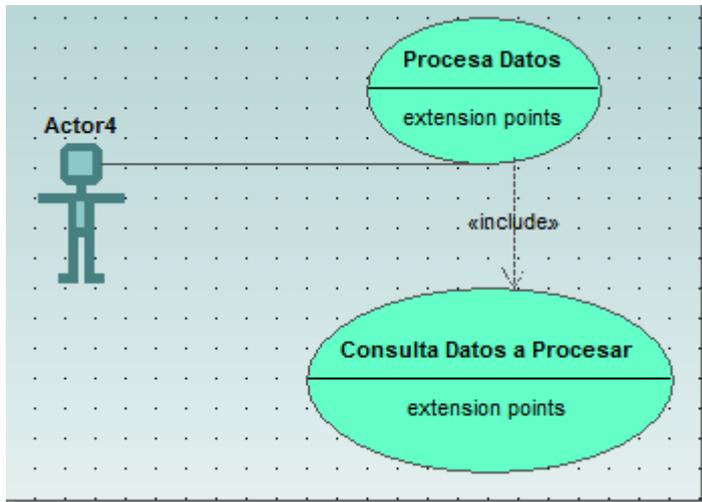
CASO DE USO: "Cargar Datos Nuevos"	
Funcionalidad	Incorporar los datos del reloj al sistema
Actores	administrador
Casos de uso relacionados	
Precondición	No debe haber datos ya cargados con las fechas indicadas.
Flujo normal	El usuario introduce las fechas para las que quiere que se tengan los datos en el sistema.
Flujo alternativo	Error en al carga. No se incorporan los datos al sistema.
Postcondición	Los datos quedan incorporados al sistema en estado cargado ó pendiente o actualizado(según indiquen los cálculos).

CASO DE USO: "Consulta Datos Cargados"	
Funcionalidad	Información de los datos que ya se han cargado en el sistema
Actores	administrador
Casos de uso relacionados	
Precondición	
Flujo normal	El usuario introduce las fechas de consulta y obtiene la información
Flujo alternativo	No hay datos para las fechas indicadas. No obtenemos datos de la consulta
Postcondición	

3.3.3. Módulo Proceso de datos.

Accesible solo al personal autorizado, donde se pueden hacer las rectificaciones correspondientes según hemos mencionado.

Gráfico de los casos de uso.



Descripción textual de los casos de uso.

CASO DE USO: "Procesa Datos"	
Funcionalidad	Modificar información del reloj para circunstancias indetectables en el sistema y el cómputo de horas de cada empleado sea correcto.
Actores	administrador
Casos de uso relacionados	
Precondición	Debe haber datos ya cargados con las fechas indicadas.
Flujo normal	El usuario introduce las fechas para las que quiere modificar datos en el sistema.
Flujo alternativo	Error en la modificación. No se incorporan los datos al sistema.
Postcondición	Los datos quedan modificados en el sistema ya no están en estado pendiente.

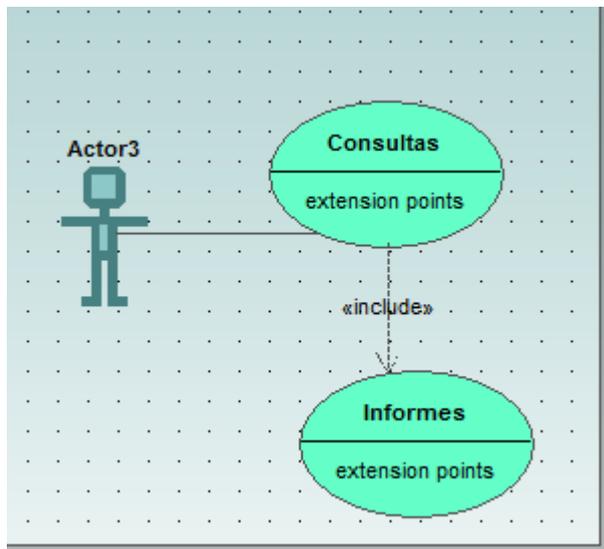
CASO DE USO: "Consulta Datos a Procesar"	
Funcionalidad	Información de los datos que ya se han cargado en el sistema
Actores	administrador
Casos de uso relacionados	
Precondición	No debe haber datos ya cargados con las fechas indicadas.
Flujo normal	El usuario introduce las fechas de consulta y obtiene la información
Flujo alternativo	
Postcondición	El sistema nos indica cuantos datos hay cargados con las fechas indicadas.

3.3.4. Módulo Consultas e Informes.

Para todo el personal que haya estado en las instalaciones donde se podrá verificar el cómputo de horas.

Donde podremos obtener información por días, semanas y demás de nuestra actividad laboral dentro de la empresa.

Gráfico de los casos de uso.



Descripción textual de los casos de uso.

CASO DE USO: "Consultas"	
Funcionalidad	Información del cómputo de horas en unas fechas determinadas
Actores	Cualquier usuario: administrador, trabajador
Casos de uso relacionados	
Precondición	
Flujo normal	El usuario introduce las fechas para las que quiere consultar.
Flujo alternativo	Para las fechas indicadas no hay datos a consultar.
Postcondición	El sistema nos informa del cómputo de horas en las fechas indicadas.

CASO DE USO: "Informes"	
Funcionalidad	Listado del cómputo de horas en unas fechas determinadas
Actores	Cualquier usuario: administrador, trabajador
Casos de uso relacionados	
Precondición	
Flujo normal	El usuario introduce las fechas para las que quiere la información.
Flujo alternativo	Para las fechas indicadas no hay datos a informar.
Postcondición	

4. Diseño.

4.1 Interfaz de usuario.

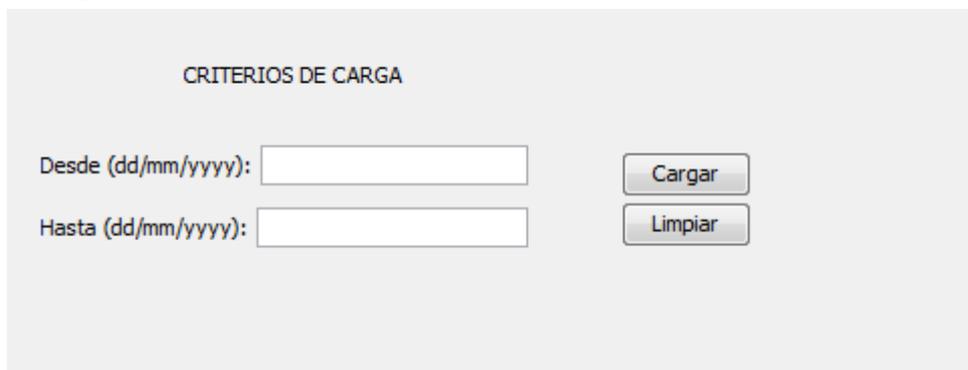
4.1.1. Módulo de login



A screenshot of a login interface. At the top center, the word "LOGIN" is displayed. Below it, there are two input fields: "Usuario:" followed by a text box, and "Contraseña:" followed by a text box. Below the password field is a button labeled "Validar".

4.1.2. Módulo de carga de datos

Cargar Datos Nuevos



A screenshot of a data loading interface. At the top center, the text "CRITERIOS DE CARGA" is displayed. Below it, there are two rows of input fields. The first row has "Desde (dd/mm/yyyy):" followed by a date input box and a "Cargar" button. The second row has "Hasta (dd/mm/yyyy):" followed by a date input box and a "Limpiar" button.

Cargar:

Los datos se han cargado correctamente con las FECHAS indicadas

Aceptar

NO hay datos a cargar con las FECHAS indicadas.

Aceptar

Consultar:

Hay XXX registros cargados con las FECHAS indicadas.

Aceptar

4.1.3. Módulo de proceso de datos

Mostrar datos para procesar:

CRITERIOS DE SELECCION

Empleados:

Desde (dd/mm/yyyy):

Hasta (dd/mm/yyyy):

Mostrar

Limpiar

Datos para procesar:

	EMPLEADO	FECHA	ENTRADA1	SALIDA	ENTRADA2	INCSALIDA	ESTADO
1	00090	20121031	08:02:23	10:37:36	10:58:16	17	A
2	00090	20121031	10:58:16	14:05:40	14:57:22	10	A
3	00090	20121031	14:57:22	16:04:13		02	A



 Para consultar marcajes originales para el empleado y fecha y decidir qué modificación haremos.

	EMPLEADO	FECHA	HORA	INCIDENCIA	ESTADO
1	EMPI	20121031	08:02:23	01	C
2	EMPI	20121031	10:37:36	17	C
3	EMPI	20121031	10:58:16	01	C
4	EMPI	20121031	14:05:40	10	C
5	EMPI	20121031	14:57:22	01	C
6	EMPI	20121031	16:04:13	02	C

 Para editar el registro y modificar para que el registro deje de estar pendiente.

4.1.4. Módulo de consultas e informes

Consultas usuario administrador

CRITERIOS DE CONSULTA

Empleados:

Desde (dd/mm/yyyy):

Hasta (dd/mm/yyyy):

Consultas usuario trabajador

CRITERIOS DE CONSULTA

Desde (dd/mm/yyyy):

Hasta (dd/mm/yyyy):

Resultados :

	Fecha	Horas
empleado: XXXXXX1	YYYY MM DD 1	N1
	YYYY MM DD 2	N2
	YYYY MM DD 3	N3
	YYYY MM DD 4	N4
	Total:	<input type="text"/>
empleado: XXXXXX2	YYYY MM DD 1	N1
	YYYY MM DD 2	N2
	YYYY MM DD 3	N3
	YYYY MM DD 4	N4
	Total:	<input type="text"/>
empleado: XXXXXX3	YYYY MM DD 1	N1
	YYYY MM DD 2	N2
	YYYY MM DD 3	N3
	YYYY MM DD 4	N4
	Total:	<input type="text"/>

4.2 Tecnología utilizada.

Entorno de desarrollo Eclipse JEE (versión para desarrolladores Web).

Eclipse Juno, versión que va muy bien con temas de spring, hibernate, jsf 2.0 y demás.

Base de datos Postgres 3.8

Apache Tomcat y Jre.

Tomcat v6.0.18 y jdk1.6.0_26. Que junto a Eclipse Juno parece que es una combinación muy estable.

Spring 3.0

Para las transacciones con la base de datos.

Jsf 2.0.

Para facilitar la creación de páginas y la navegación.

Plugging para hibernate.

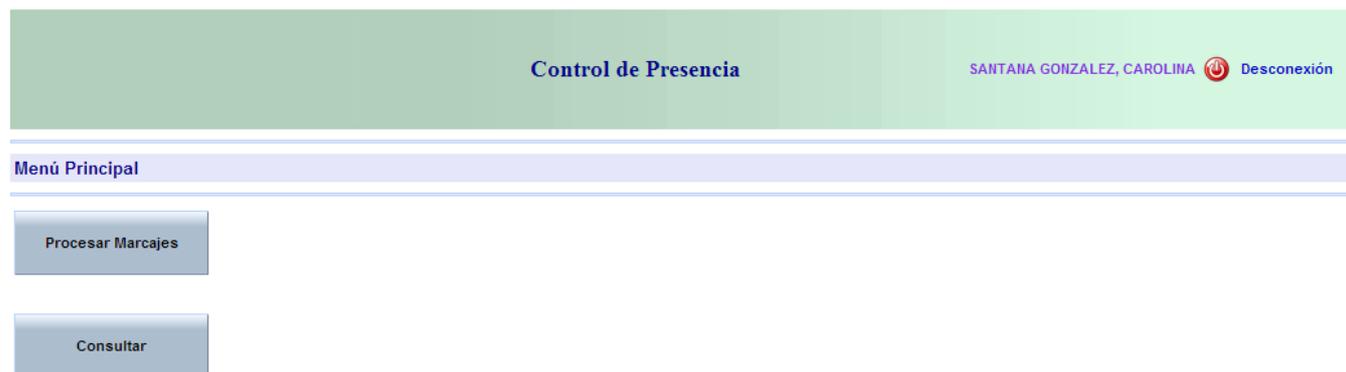
Una vez instalado el plugging se configura la consola hibernate, para generar correctamente el mapeo de las tablas.

4.3 Imágenes capturadas en funcionamiento.

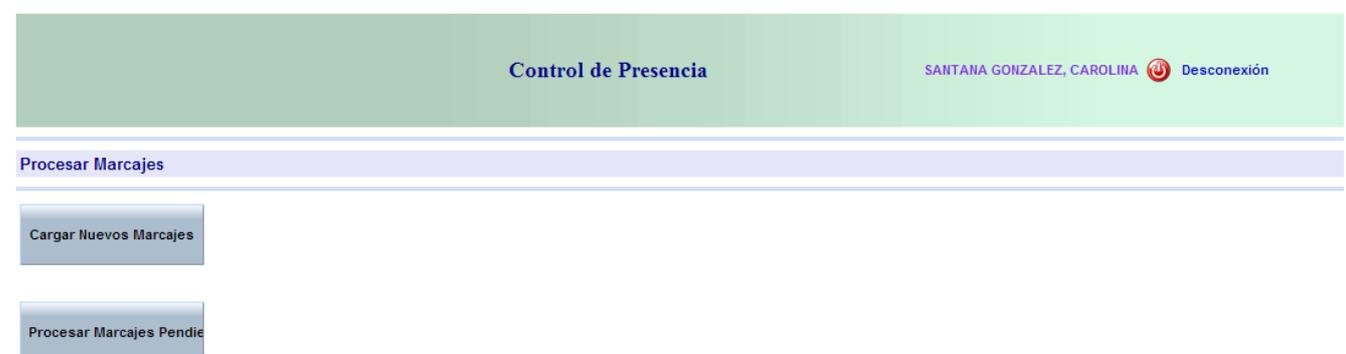
LOGIN



MENU PRINCIPAL



PROCESAR MARCAJES (solo administradores)



Procesar Marcajes. **CARGAR NUEVOS MARCAJES**

Control de Presencia SANTANA GONZALEZ, CAROLINA  Desconexión

Cargar Nuevos Marcajes

Criterios de Carga

Desde (dd/mm/yyyy)  Hasta (dd/mm/yyyy) 

Resultados CARGAR NUEVOS MARCAJES

Control de Presencia SANTANA GONZALEZ, CAROLINA  Desconexión

Cargar Nuevos Marcajes

Criterios de Carga

Desde (dd/mm/yyyy) 08/10/2012  Hasta (dd/mm/yyyy) 

 **El TRAMO de FECHAS no es correcto**

Control de Presencia SANTANA GONZALEZ, CAROLINA  Desconexión

Cargar Nuevos Marcajes

Criterios de Carga

Desde (dd/mm/yyyy) 12/12/2012  Hasta (dd/mm/yyyy) 13/12/2012 

 **NO HAY Datos a cargar con este TRAMO de FECHAS**

Control de Presencia SANTANA GONZALEZ, CAROLINA  Desconexión

Cargar Nuevos Marcajes

Criterios de Carga

Desde (dd/mm/yyyy) 03/09/2012  Hasta (dd/mm/yyyy) 03/12/2012 

 **Ya HAY Dias cargados en este TRAMO de FECHAS**

Resultados CARGAR NUEVOS MARCAJES

Control de Presencia SANTANA GONZALEZ, CAROLINA  Desconexión

Cargar Nuevos Marcajes

Criterios de Carga

Desde (dd/mm/yyyy) 08/10/2012  Hasta (dd/mm/yyyy) 08/10/2012 

i Marcajes CARGADOS y GENERADOS correctamente en el TRAMO de FECHAS.

Procesar Marcajes. PROCESAR MARCAJES PENDIENTES

Control de Presencia SANTANA GONZALEZ, CAROLINA  Desconexión

Procesar Marcajes Pendientes

Criterios de Selección

Desde (dd/mm/yyyy) 01/05/2009  Hasta (dd/mm/yyyy) 08/05/2009 

Código de empleado

Consultas de Pendientes											
Empleado	Fecha	Entrada1	Salida	Entrada2	IncSalida	Estado	Mpresencia	Mdesayuno	Ver	Edi	Add
usu01	20090504	07:33:1207:33:12	15:17:40		02	A	465	0	?		A
usu01	20090505	07:35:3107:35:31	11:08:49	11:48:20	17	A	213	39	?		A
usu01	20090505	11:48:2011:48:20	15:29:00		02	A	221	0	?		A
usu01	20090506		15:03:59		02	P	0	0	?		P
usu01	20090507	07:33:3307:33:33	15:05:16		02	A	451	0	?		A
usu01	20090508	07:45:5207:45:52	12:03:11	12:32:59	17	A	257	30	?		A
usu01	20090508	12:32:5912:32:59	15:04:09		02	A	151	0	?		A

Resultados PROCESAR MARCAJES PENDIENTES. Consultar marcajes

Control de Presencia

SANTANA GONZALEZ, CAROLINA Desconexión

Procesar Marcajes Pendientes

Criterios de Selección

Desde (dd/mm/yyyy) 01/05/2009 Hasta (dd/mm/yyyy) 08/05/2009

Código de empleado

Aceptar

Limpiar

Relación de Marcajes Originales

Consultas de marcajes originales				
Empleado	Fecha	Hora	Incidencia	Estado
usu01	20090504	07:33:12	01	C
usu01	20090504	15:17:40	02	C

Consultas de Pendientes

Empleado	Fecha	Entrada1	Salida	Entra2	Inc.Sal	Est	Mpre	Mdes	V	E
usu01	20090504	07:33:12	15:17:40		02	A	470	0	?	
usu01	20090505	07:35:31	11:08:49	11:48:20	17	A	213	39	?	
usu01	20090505	11:48:20	15:29:00		02	A	221	0	?	
usu01	20090506		15:03:59		02	P	0	0	?	
usu01	20090507	07:33:33	15:05:16		02	A	451	0	?	
usu01	20090508	07:45:52	12:03:11	12:32:59	17	A	257	30	?	
usu01	20090508	12:32:59	15:04:09		02	A	151	0	?	

Resultados PROCESAR MARCAJES PENDIENTES. Editar marcajes.

Control de Presencia SANTANA GONZALEZ, CAROLINA

Procesar Marcajes Pendientes

Criterios de Selección

Desde (dd/mm/yyyy) 01/05/2009 Hasta (dd/mm/yyyy) 05/05/2009

Código de empleado

Consultas de Pendientes										
Empleado	Fecha	Entrada1	Salida	Entra2	IncSal	Est	Mpre	Mdes	V	E
usu01	20090504	07:33:12	15:17:40		02	A	466	0	?	
usu01	20090505	07:35:31	11:08:49	11:48:20	17	A	213	39	?	
usu01	20090505	11:48:20	15:29:00		02	A	221	0	?	

Mpre = 466

Control de Presencia SANTANA GONZALEZ, CAROLINA

Procesar Marcajes Pendientes

Criterios de Selección

Desde (dd/mm/yyyy) 01/05/2009 Hasta (dd/mm/yyyy) 05/05/2009

Código de empleado

Consultas de Pendientes										
Empleado	Fecha	Entrada1	Salida	Entra2	IncSal	Est	Mpre	Mdes	V	E
usu01	20090504	07:33:12	15:17:40		02	A	470	0	?	
usu01	20090505	07:35:31	11:08:49	11:48:20	17	A	213	39	?	
usu01	20090505	11:48:20	15:29:00		02	A	221	0	?	

Mpre = 470 (aquí podemos aceptar o cancelar la modif)

Si volvemos a realizar la búsqueda en base de datos veremos que el nuevo dato se ha grabado. En caso de cancelar veríamos el dato original.

Control de Presencia SANTANA GONZALEZ, CAROLINA

Procesar Marcajes Pendientes

Criterios de Selección

Desde (dd/mm/yyyy) 04/05/2009 Hasta (dd/mm/yyyy) 04/05/2009

Código de empleado

Consultas de Pendientes										
Empleado	Fecha	Entrada1	Salida	Entra2	IncSal	Est	Mpre	Mdes	V	E
usu01	20090504	07:33:12	15:17:40		02	A	470	0	?	

CONSULTAR (todos tipo de usuarios)

Control de Presencia SANTANA GONZALEZ, CAROLINA Desconexión

Consultar Marcajes

Criterios de Consulta

Desde (dd/mm/yyyy) Hasta (dd/mm/yyyy)

Código de empleado

Consultas

Empleado	Fecha	Minutos
No se encontraron registros.		

Consultar todos los empleados (sólo administradores)

Consulta únicamente su código de empleado (trabajadores)

Resultado CONSULTAS con datos de prueba

Control de Presencia SANTANA GONZALEZ, CAROLINA Desconexión

Consultar Marcajes

Criterios de Consulta

Desde (dd/mm/yyyy) 01/01/2012 Hasta (dd/mm/yyyy) 15/01/2012

Código de empleado

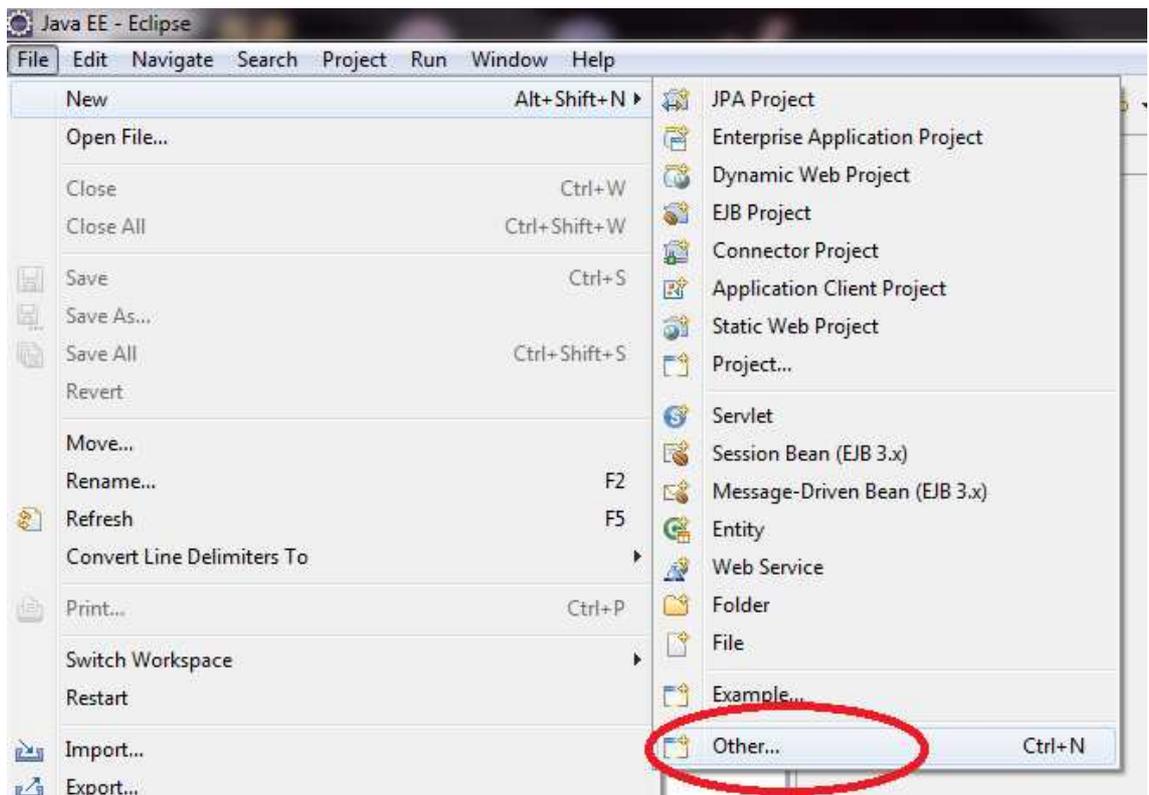
Empleado	Fecha	Minutos
usu01	20120101	20
	20120102	30
	20120103	40
usu01	Total:	90
usu02	20120101	10
	20120110	20
usu02	Total:	30
usu03	20120109	15
usu03	Total:	15

5. Manuales.

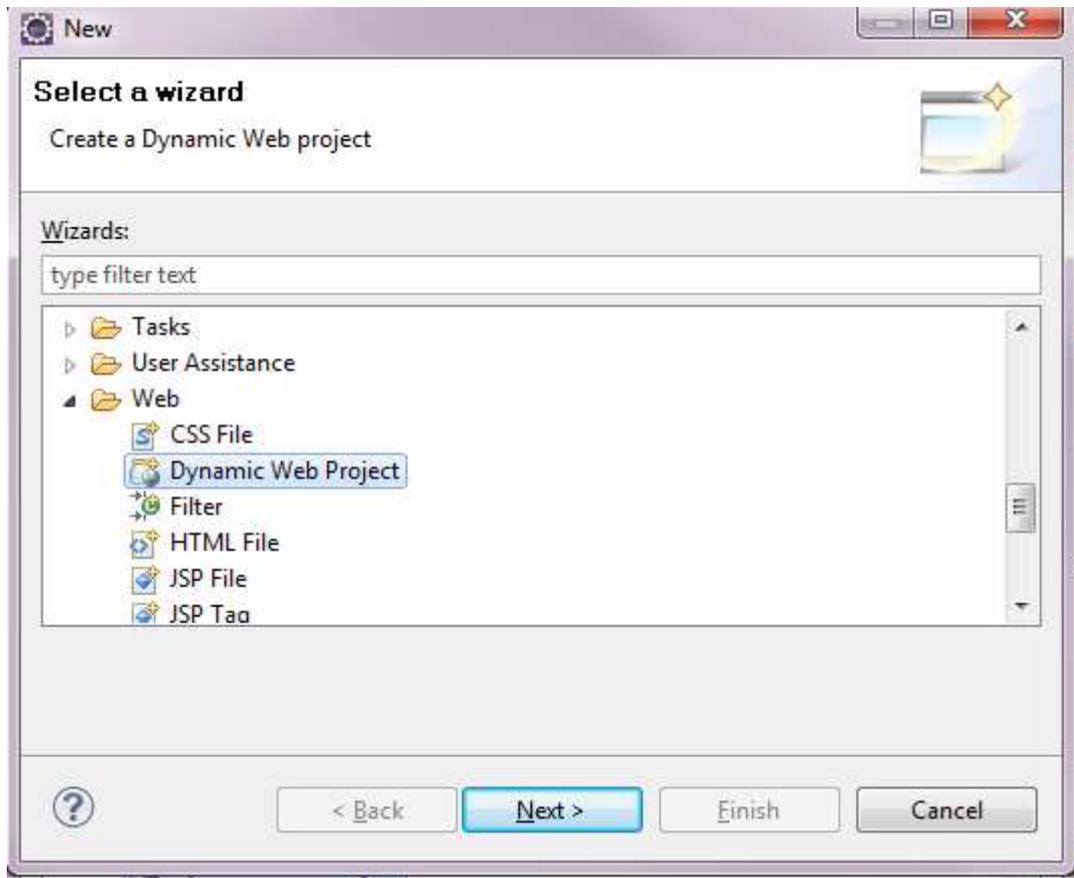
5.1 Guía para crear el proyecto.

El proyecto se debe crear con unas ciertas características para se adecue a la tecnología que queremos utilizar.

1º Empezamos en Eclipse creando el proyecto web con las opciones: **File, New, Other**



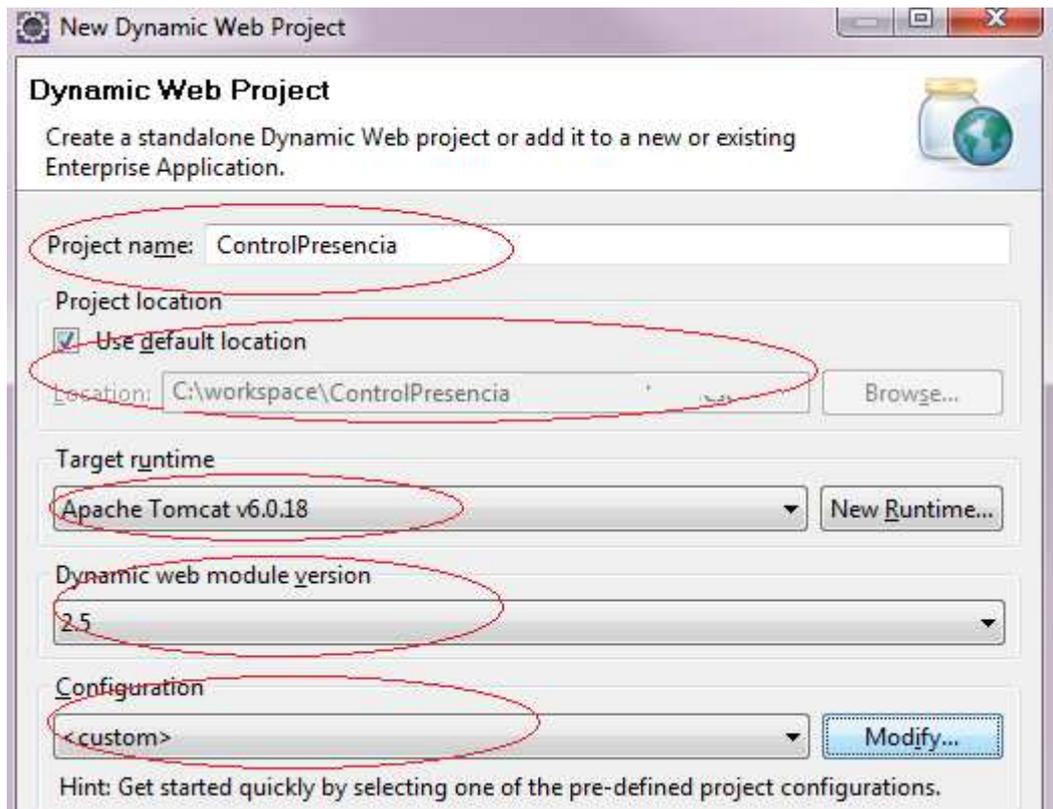
2º Con Wizard personalizamos el proyecto: elegimos **Dynamic Web Project**, **Next**



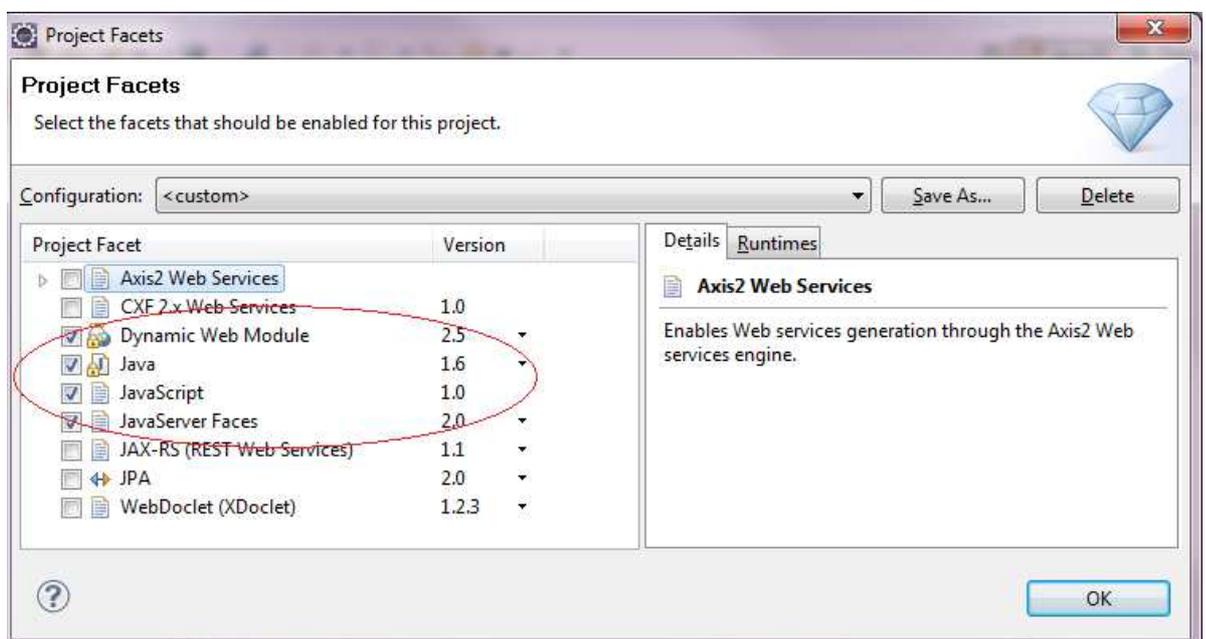
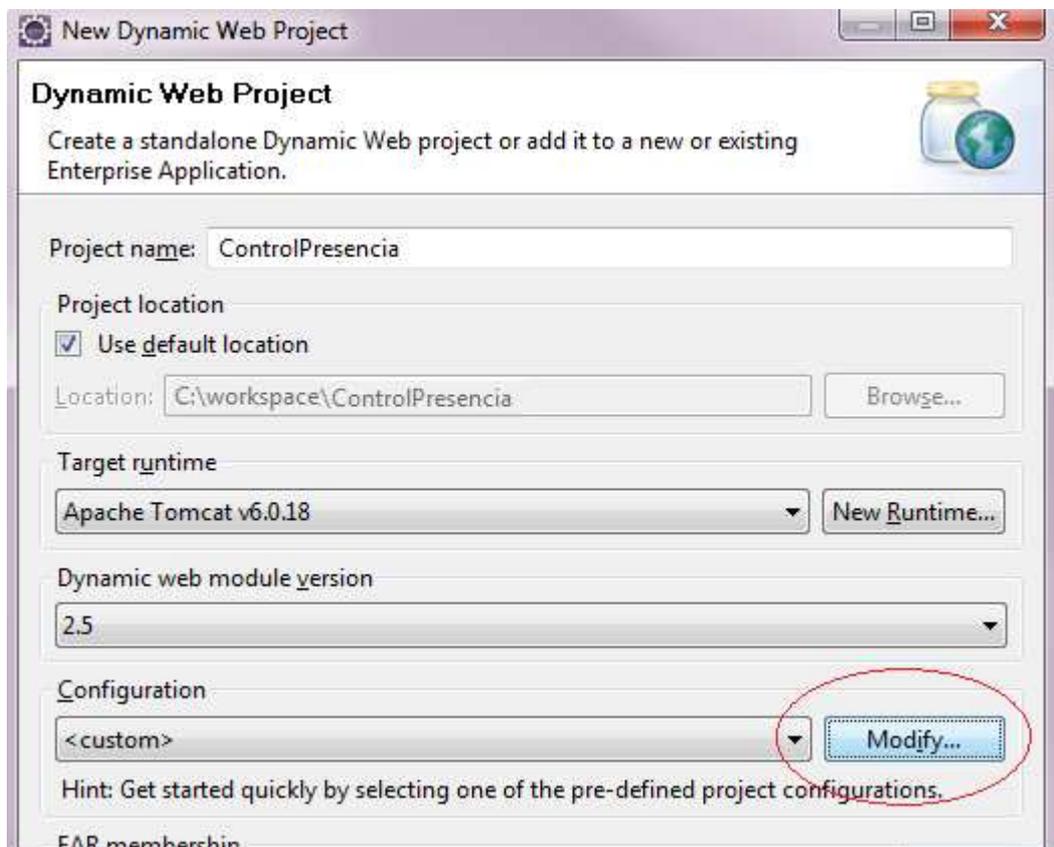
3º Elegimos el **Nombre**, **Localización** (WorkSpace), **Runtime**(Tomcat), **Dynamic web module versión** y **Configuración**.

Muy importantes! los valores que se indican en la figura ya que trabajaremos JSF 2.0, HIBERNATE Y demás...

Dynamic web module versión: 2.5 → por la versión del servlet 2.5



4º Customizar la Configuración: **Modify**, verificar las opciones de la imagen, **OK**.



5º **FINISH**

5.2 Guía para configurar spring e instalar, configurar y utilizar la consola de Hibernate.

5.2.1 Configurar spring.xml hibernate.cfg.xml

spring.xml

- 1º Debe alojarse dentro de WEB-INF
- 2º Debemos indicar los parámetros de la BD utilizada.

Necesitamos **Nombre** de BD ; **IP**;
 Usuario y Clave.

```
<bean id="dataSource" class=
    "org.springframework.jdbc.datasource.DriverManagerDataSource">
  <property name="driverClassName" value="org.postgresql.Driver" />
  <property name="url" value="jdbc:postgresql:cpresen" />
  <property name="username" value="postgres" />
  <property name="password" value="*****" />
</bean>

<!-- integración entre Spring e Hibernate -->

<bean id="sessionFactory" class=
  "org.springframework.orm.hibernate3.annotation.AnnotationSessionFactoryBean">

  <property name="dataSource"><ref local="dataSource" /></property>
  <property name="configLocation" value=
    "/WEB-INF/hibernatePostgreCpresen.cfg.xml"></property>
  <property name="annotatedPackages">
    <list>
      <value>org.grecasa.intranet.cPresencia.dto</value>
    </list>
  </property>
  <property name="annotatedClasses">
    <list>
      <value>org.grecasa.intranet.cPresencia.dto.Presencargo</value>
      <value>org.grecasa.intranet.cPresencia.dto.Cpresencia</value>
      <value>org.grecasa.intranet.cPresencia.dto.Tmpcpresencia</value>

      <value>org.grecasa.intranet.cPresencia.dto.Cpempleados</value>
    </list>
  </property>
</bean>
```

- 3º Integrar *hibernate - spring*.
 Crear el fichero ***hibernate.cfg.xml*** y referenciarlo en el propio *spring.xml*.

```
<hibernate-configuration>
<session-factory name="">
  <property name=
    "hibernate.connection.driver_class">org.postgresql.Driver
  </property>
  <property name="hibernate.connection.url">jdbc:postgresql:cpresen
  </property>
  <property name="hibernate.connection.username">postgres</property>
  <property name="hibernate.connection.password">*****</property>
  <property name=
    "hibernate.dialect">org.hibernate.dialect.PostgreSQLDialect
  </property>
  <property name="hibernate.cglib.use_reflection_optimizer">>true
  </property>
  <property name="hibernate.show_sql">>false</property>
  <property name="hibernate.connection.pool_size">1</property>
</session-factory>
</hibernate-configuration>
```

Crear el fichero ***hibernate.reveng.xml***
 (guia para el mapeo hibernate)

```
<hibernate-reverse-engineering>
  <type-mapping>
    sql-type jdbc-type="DATE" hibernate-type="timestamp"></sql-type>
  </type-mapping>
  <table-filter match-catalog="cpresen" match-schema="dbo"
    match-name="presencargo"/>
  <table-filter match-catalog="cpresen" match-schema="dbo"
    match-name="cpresencia"/>
  <table-filter match-catalog="cpresen" match-schema="dbo"
    match-name="tmpcpresencia"/>
  <table-filter match-catalog="cpresen" match-schema="dbo"
    match-name="cpempleados"/>
</hibernate-reverse-engineering>
```

5.2.2 Instalación del software hibernate-tools

1º Debemos buscar el plugin hibernate-tools para Eclipse desde la url:

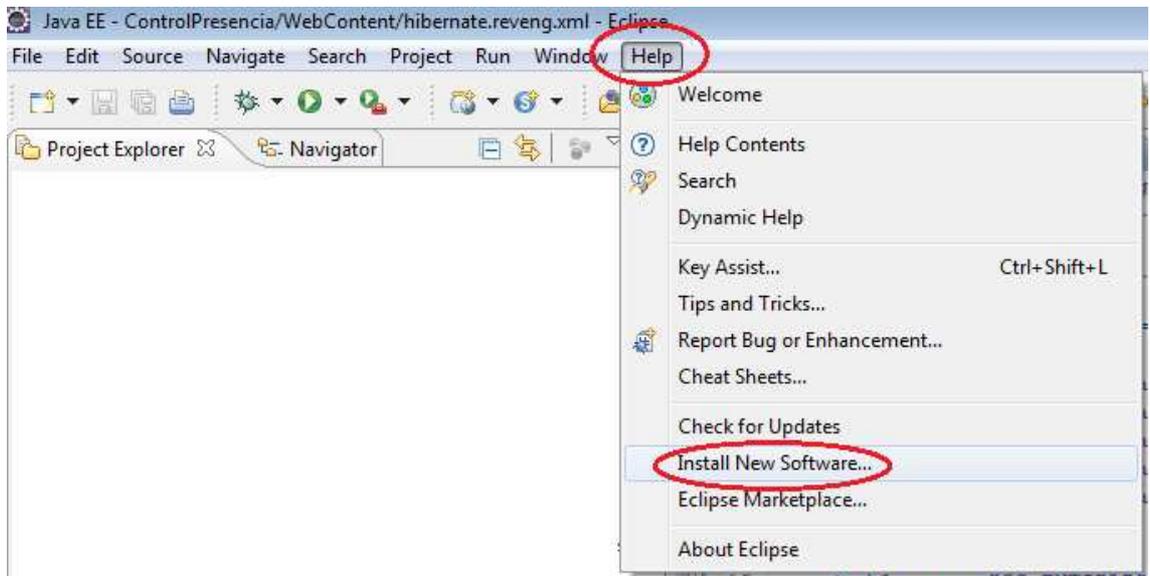
<http://www.mkyong.com/hibernate/how-to-install-hibernate-tools-in-eclipse-ide/>

2º En nuestro caso y para Eclipse Juno, el plugin está disponible en la url:

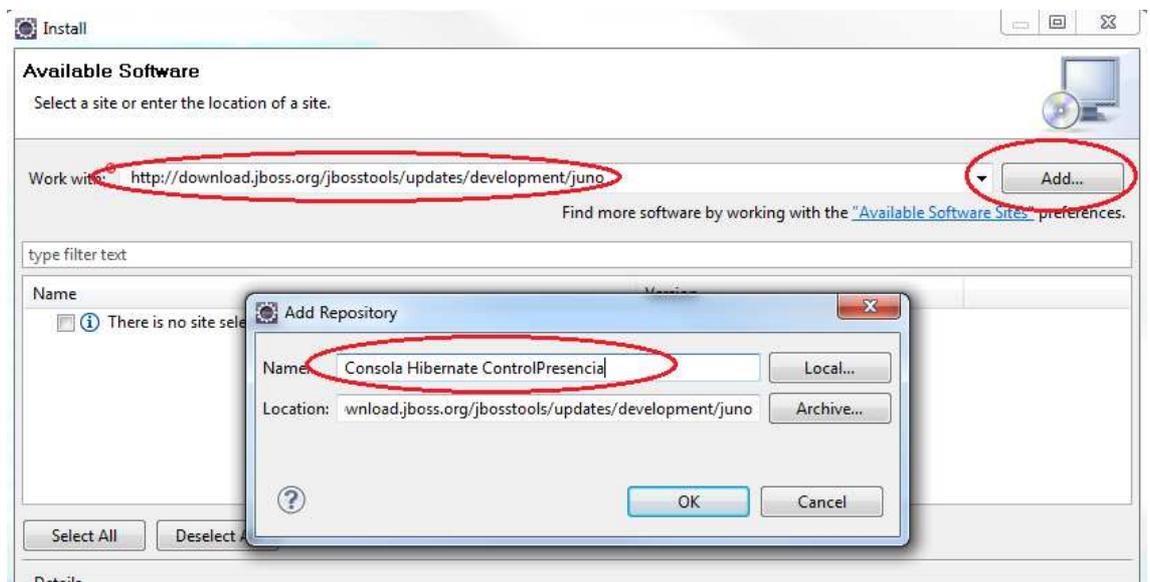
<http://download.jboss.org/jbosstools/updates/development/juno/>

3º Instalar el nuevo software:

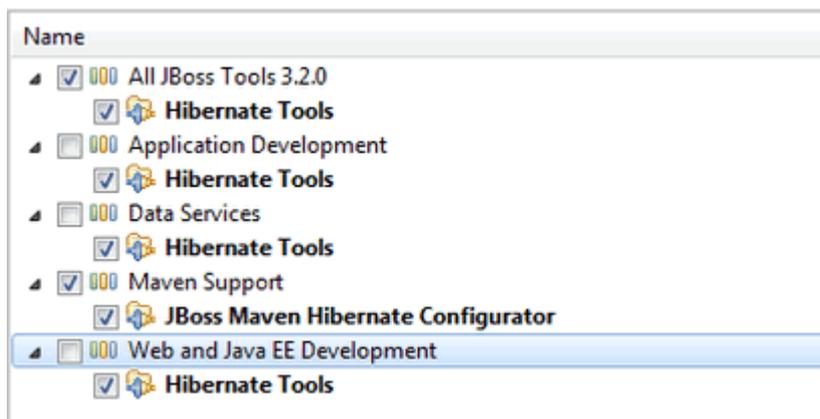
1- **Seleccionar** en el eclipse la opciónes **Help**, **Install New Software**.



2- Indicar la **url del nuevo software**, **Add** y dar un **nombre a la consola**



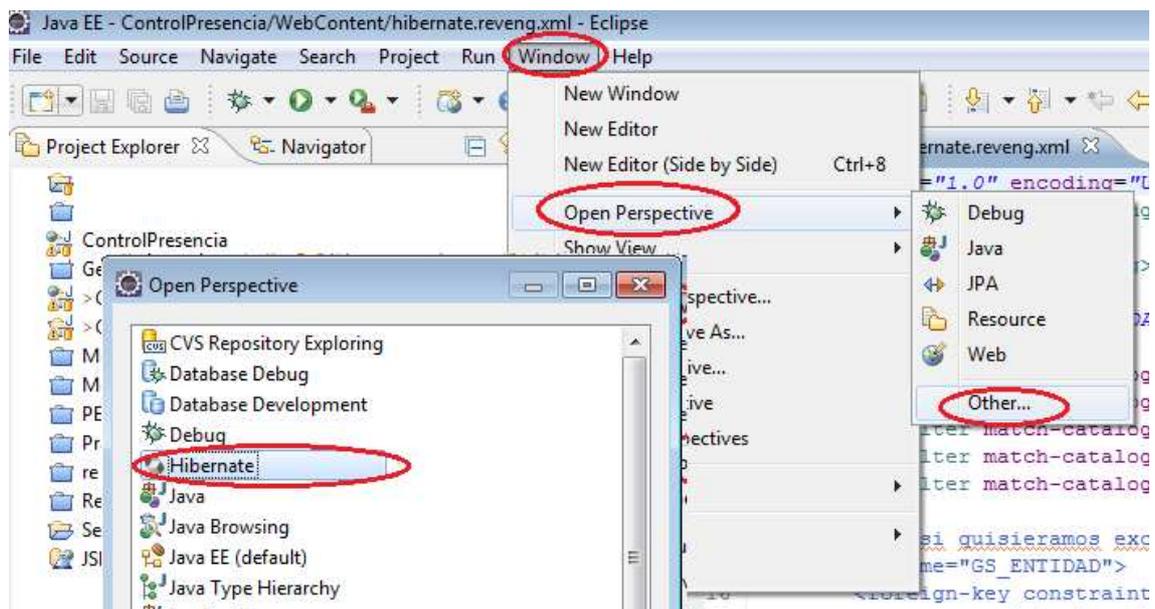
3- Ir seleccionar las hibernate tools y solo las necesarias tal y como se ve en la imagen.



4- Ir avanzando con **Next**, **Aceptar la licencia** y **Finish**

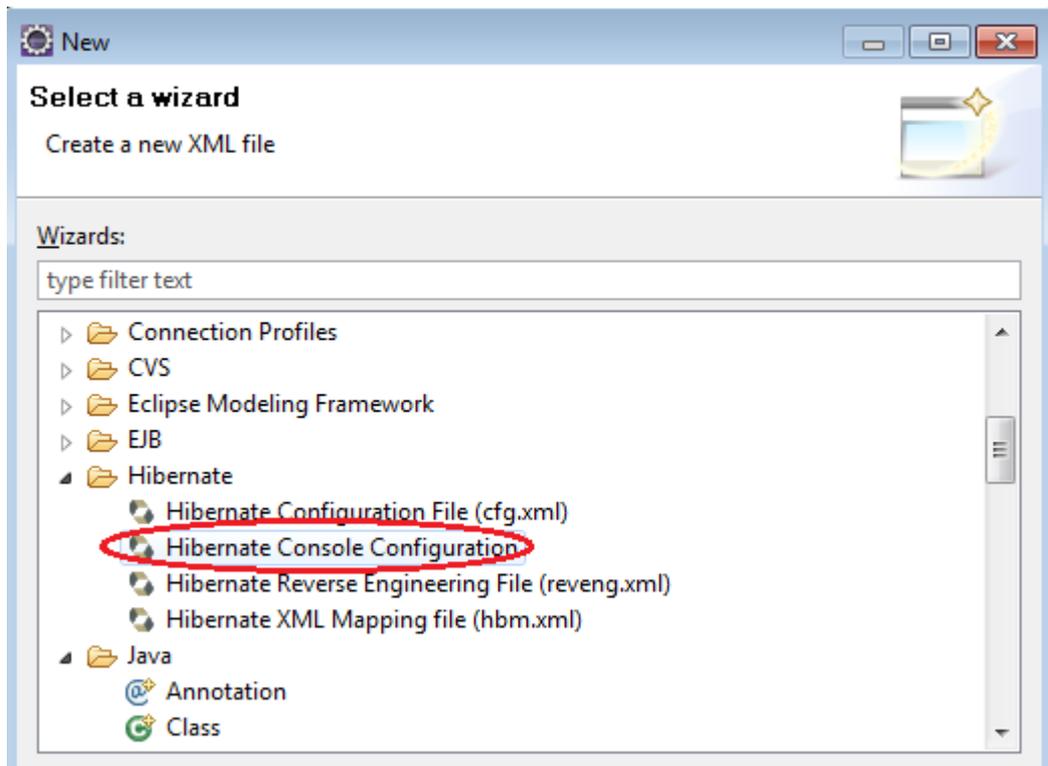
5- Reiniciar el Eclipse.

6- Asegurar que el proceso se ha completado accediendo a las perspectivas y comprobar que tenemos la de Hibernate.



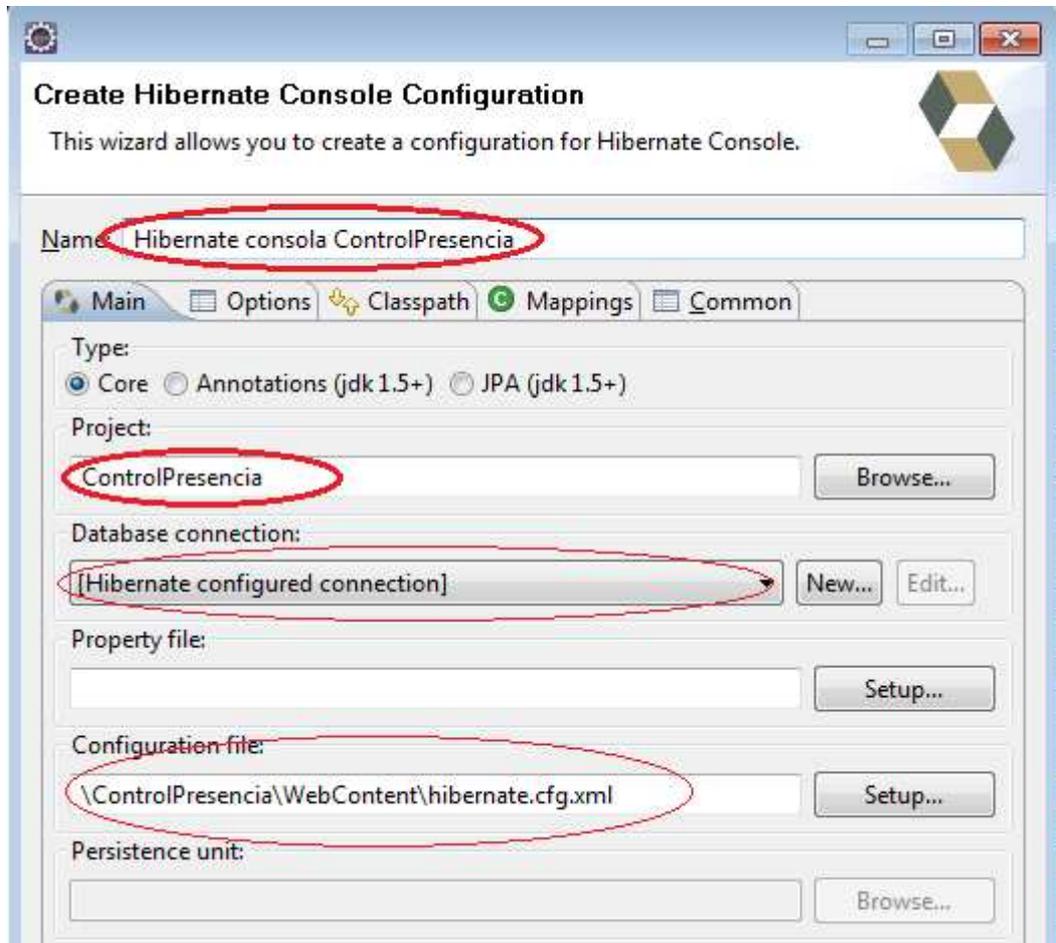
5.2.3 Configurar y utilizar la consola de hibernate.

- 1- Sobre nuestro proyecto: boton derecho, New , Other, Hibernate, Hibernate Console configuration

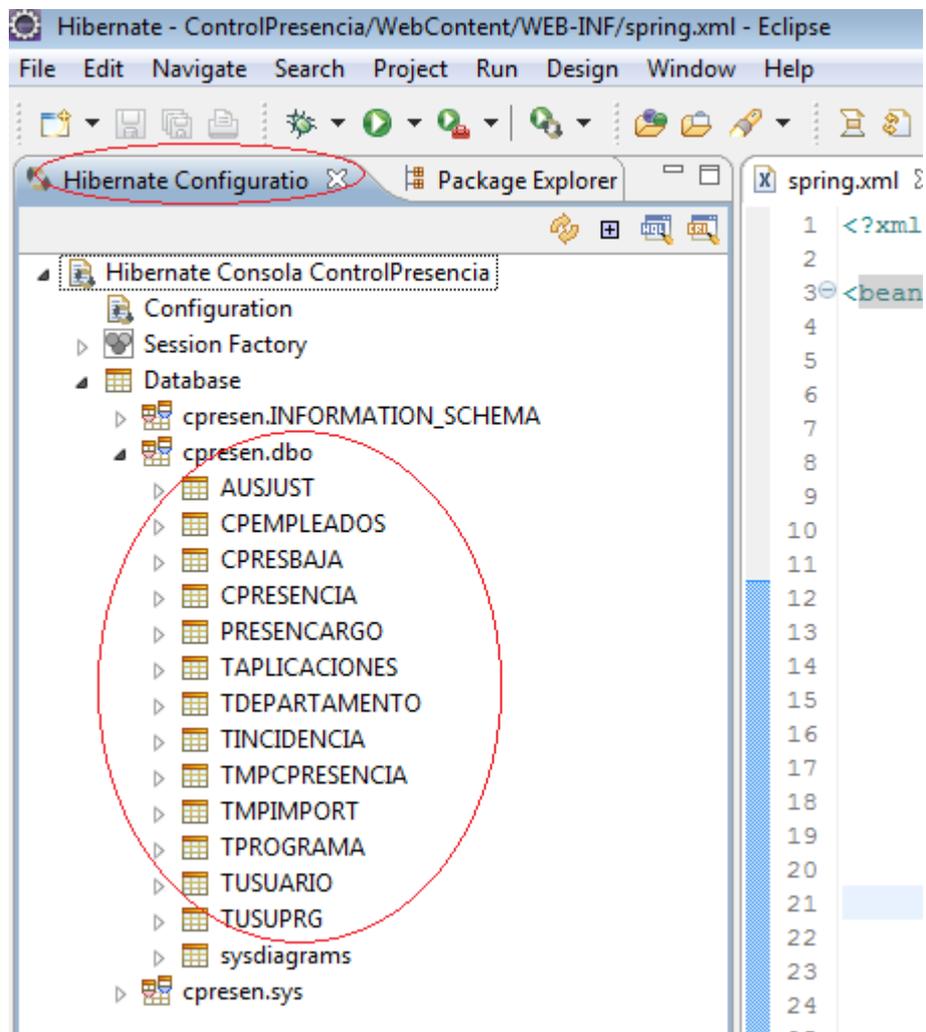


2- Indicamos:

nombre de consola, nombre del **proyecto** y **ruta fichero de config** hibernate

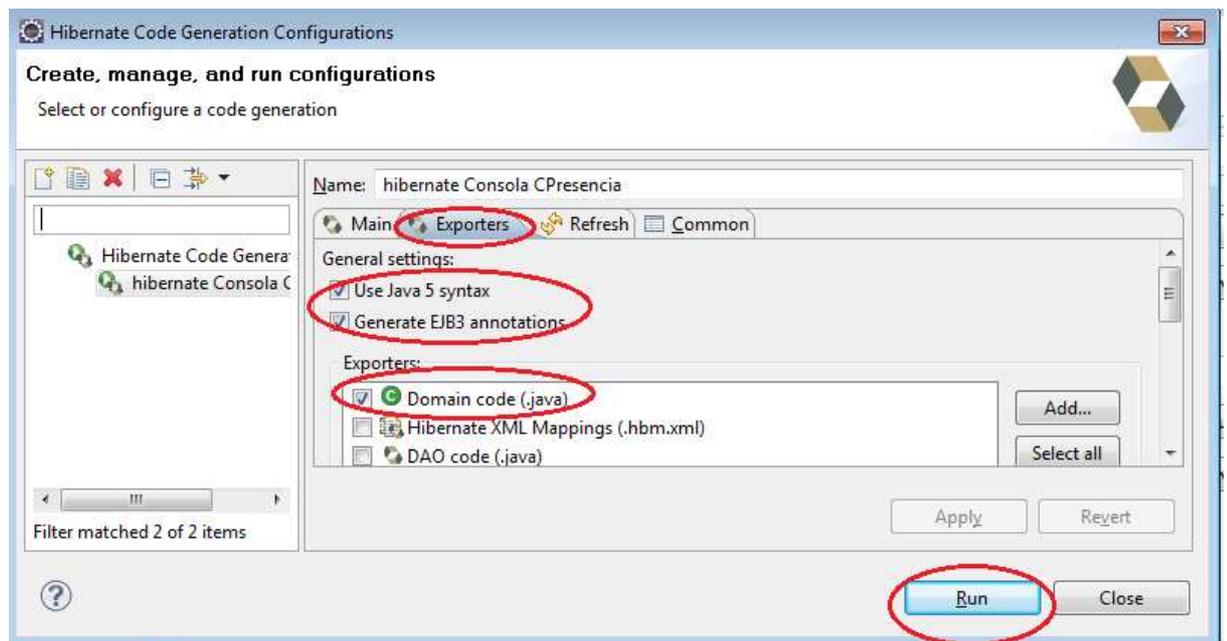
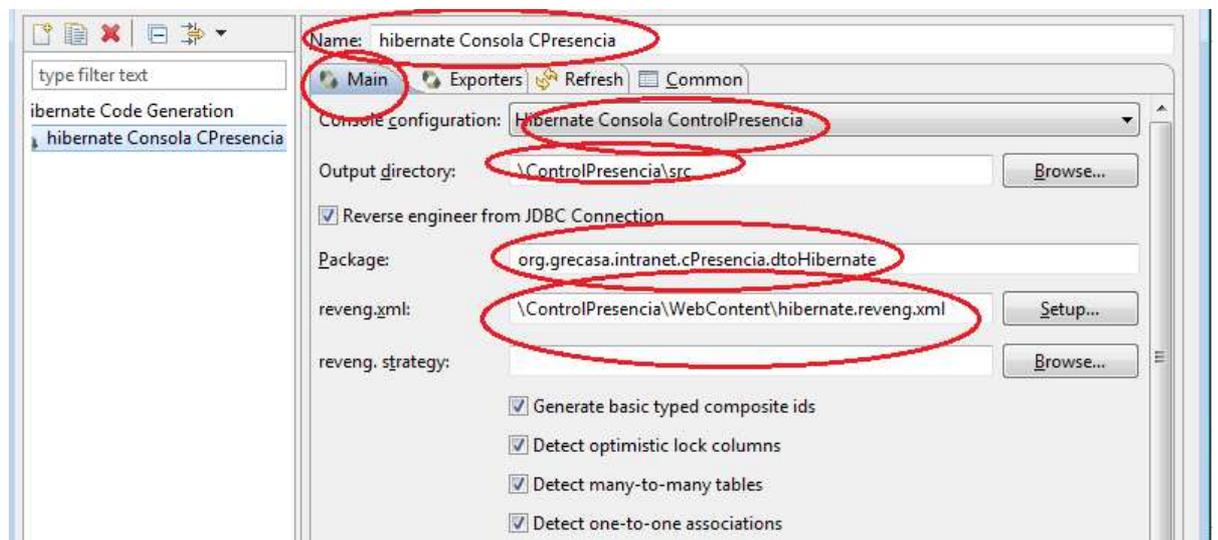
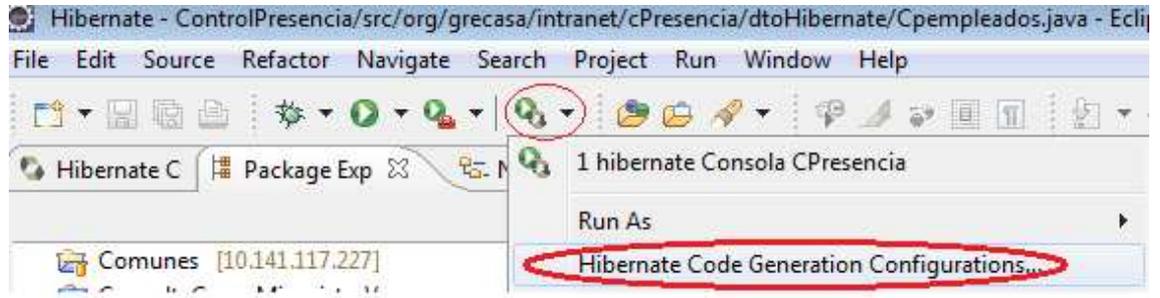


- 3- Comprobamos en el entorno que hibernate conecta correctamente con la BD.

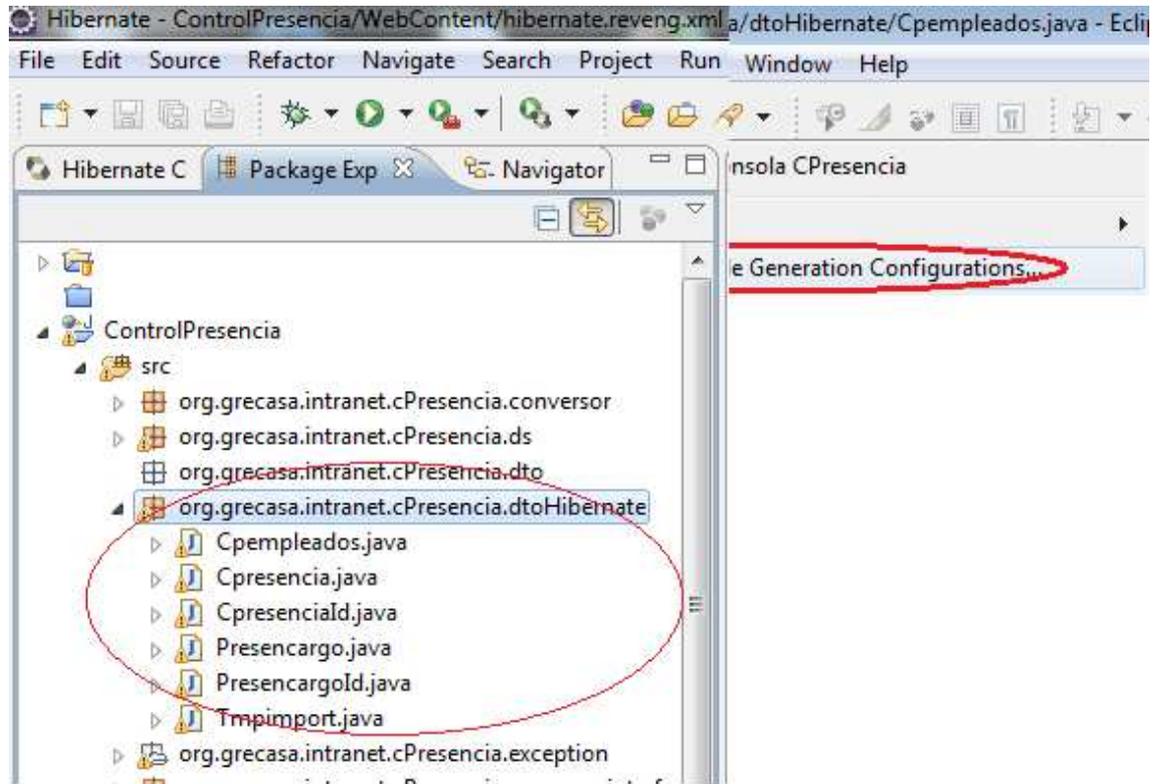


- 4- Tenemos que **crear un nuevo paquete** para alojar el mapeo de hibernate; en nuestro caso: **org.grecasa.intranet.cPresencia.dtoHibernate**

5- Utilizando la consola:



- 6- Para asegurarnos de que ha funcionado el mapeo de hibernate tenemos que comprobar que el nuevo paquete creado anteriormente (`org.grecasa.intranet.cPresencia.dtoHibernate`) contiene las nuevas dto's de las tablas que se indicaron en el fichero `hibernate.reveng.xml`



6. Conclusiones.

La intención de hacer este proyecto ha sido crear una aplicación útil tanto en lo conceptual como en lo técnico.

Conceptualmente hablando, que realmente sirviera para mejorar la gestión del control de presencia y técnicamente, que propiciara la plataforma para acercarnos a la tecnología utilizada para aprender y ampliar conocimientos.

En el tiempo que he estado desarrollando esta aplicación se han ido dando una serie de circunstancias en la empresa para la que trabajo, cambios horarios de la jornada, necesidades fuera de las dependencias de la misma y demás eventualidades que me dejan patente la utilidad y necesidad de un producto de este tipo.

Por otro lado, ha sido todo un reto comprender y manejar esta tecnología que era totalmente nueva para mí.

¡Y esto ha sido solo el principio!