

**EGRESADOS WEB Y
GESTION PUBLICA HOJA DE VIDA EGRESADOS DE
LA FUNDACIÓN UNIVERSITARIA DEL ÁREA ANDINA**

JORGE ANDRÉS CAÑAS VARGAS

WALTER JULIÁN BERNAL SUÁREZ

**CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE TECNOLOGÍA EN INFORMÁTICA
BOGOTÁ, D. C.
2007**

**EGRESADOS WEB Y
GESTION PUBLICA HOJA DE VIDA EGRESADOS DE
LA FUNDACIÓN UNIVERSITARIA DEL ÁREA ANDINA**

**JORGE ANDRÉS CAÑAS VARGAS
WALTER JULIÁN BERNAL SUÁREZ**

LIC. DAIRO ANTONIO MUÑOS

Asesor

**CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE TECNOLOGÍA EN INFORMÁTICA
BOGOTÁ, D. C.
2007**

Nota de Aceptación

Firma presidente del jurado

Firma del jurado

Firma del jurado

Bogotá, 03 / 08 / 07

AGRADECIMIENTOS

Agradecemos a la Corporación Universitaria Minuto de Dios por impartir correctamente su cátedra y brindar los conocimientos necesarios que nos facilitaron la realización de nuestra pasantía; igualmente a la Fundación Universitaria del Área Andina por aceptar estudiantes que deseen poner en práctica sus conocimientos como pasantes, y poder adquirir una mayor experiencia en el área.

Agradecemos especialmente a las divisiones: Centro de Proyección Social y Egresados, Informática Educativa y Sistemas e Informática de la Fundación Universitaria del Área Andina por colaborarnos en este proceso de práctica – aprendizaje.

CONTENIDO

	Pág.
INTRODUCCIÓN	10
1.1. TEMA	11
1.2. TÍTULO	12
1.3. PLANTEAMIENTO DEL PROBLEMA	13
1.4. DESCRIPCIÓN DEL PROBLEMA	14
1.5. JUSTIFICACIÓN	15
1.6. OBJETIVOS	16
1.6.1. OBJETIVO GENERAL	16
1.6.2. OBJETIVOS ESPECÍFICOS	16
1.7. METODOLOGÍA DE INVESTIGACIÓN	17
1.8. LÍNEA DE INVESTIGACIÓN	18
2. MARCO REFERENCIAL	20
2.1. ESTADO DEL ARTE	20
2.2. FUNDAMENTOS TEÓRICOS	23
2.3. REFERENCIA ORGANIZACIONAL	27
2.3.1. ANTECEDENTES	27
2.3.2. MISIÓN	27
2.3.3. VISIÓN	27
2.3.4. ESTRUCTURA ORGANIZACIONAL	28
3. INGENIERÍA DEL PROYECTO	29
3.1.1. METODOLOGÍA Y MODELO DE DESARROLLO	29
4. ANÁLISIS	31
4.1. DEFINICIÓN DEL SISTEMA ACTUAL	31
4.2. DESCRIPCIÓN DETALLADA DEL SISTEMA ACTUAL	32
4.3. DESCRIPCIÓN DETALLADA DE CADA PROCESO	33
4.4. DIAGRAMA DE ENTRADA SALIDA	36

5. DISEÑO	37
5.1. DICCIONARIO DE DATOS	37
5.2. MODELO ENTIDAD RELACIÓN	46
5.3. CONCEPTUALIZACIÓN DEL SISTEMA PROPUESTO	47
5.4. DISEÑO DEL SISTEMA PROPUESTO	48
5.5. DIAGRAMA DE CADA PROCESO	50
6. DESARROLLO	51
6.1. ESPECIFICACIONES TÉCNICAS	51
6.1.1. SOFTWARE	51
6.1.2. HARDWARE	51
6.2. ESTRUCTURA DEL CÓDIGO FUENTE	52
7. PRUEBAS	54
8. CRONOGRAMA	55
9. GLOSARIO	57
10. CONCLUSIONES	59
BIBLIOGRAFÍA	60
ANEXOS	
10.1. MANUAL DEL SISTEMA	
10.2. MANUAL DEL USUARIO	
10.3. MANUAL DE CONFIGURACIÓN	

LISTAS DE FIGURAS

	Pág.
Diagrama 1. “Estructura Organizacional”	28
Diagrama 2. “Modelo en Cascada”	29
Diagrama 3. “Entrada y Salida de Datos”	36
Diagrama 4. “Modelo Entidad - Relación”	46
Diagrama 5. “Diseño Gestión Pública Hoja de vida de Egresado 1”	48
Diagrama 6. ““Diseño Gestión Pública Hoja de vida de Egresado 2”	48
Diagrama 7. ““Diseño Gestión Pública Hoja de vida de Egresado 3”	49
Diagrama 8. “Diagrama de Ingreso”	50
Diagrama 9. “Diagramas de Modificar/Actualizar”	50
Diagrama 10. “Estructura del Código – Egresados Web”	52
Diagrama 11. “Estructura del Código – Gestión Pública”	53

LISTA DE ANEXOS

	Pág.
Tabla 1. "Egresados Web"	37
Tabla 2. "Gestión Pública Hoja de vida de Egresado"	42

I. INTRODUCCIÓN

Con la rápida evolución de los sistemas de almacenamiento de la información y de las comunicaciones, con el fin de facilitar y dinamizar procesos, se ha generado hoy en día, una serie de páginas y/o Portales Web, que ofrecen una serie de productos y servicios para sus usuarios.

Estos sistemas de Información nacen de la necesidad de ofrecer a los usuarios, servicios de forma fácil e integrada, el acceso a sus productos y servicios, con el fin de satisfacer sus necesidades.

Dada la importancia de la figura del egresado como parte de la comunidad universitaria, se ratificó la necesidad de desarrollar un sistema de información con acceso Web que permitiese, de manera centralizada y consolidada en el Sistema de Información SINU, la actualización y modificación de los datos de un egresado contenidos allí con el fin de obtener mediante consultas la información para las diferentes entidades del estado como el MEN (Ministerio de Educación Nacional) con sus sistemas SNIES (Sistema Nacional de Información de la Educación Superior), SPADIES (Sistema de Prevención y Análisis de la Deserción en Las Instituciones de Educación Superior) y demás IES (Instituciones de educación Superior), así como las internas de la Fundación. Lo anterior, buscando fortalecer los lazos entre egresados y la Fundación por medio de la vía de comunicación e interacción con el sistema de Información SINU.

Por lo tanto, el presente documento contiene una serie de actividades y proyectos que se realizaron durante la pasantía en la Fundación Universitaria del Área Andina, colaborando al Centro de Proyección Social y Egresados, con asesoría de Informática Educativa y posteriormente con la División de Sistemas e Informática, como lo son Egresados Web y Gestión Pública Hoja de Vida de Egresados de la Fundación Universitaria del Área Andina, el primero desarrollado e implementado en el IV Encuentro de Egresados de la Fundación Universitaria del Área Andina; y el segundo, una recopilación de las diferentes bases de datos, creando una única completa y de fácil acceso.

1.1 TEMA

Los proyectos que se desarrollaron contienen bases datos de tipo relacional, debido a que se almacena un gran número de datos, los cuales poseen una serie de atributos que se relacionan y/o encuentran en un determinado número de tablas; en ambos proyectos se trabaja con la información de terceros, como lo son los egresados de la Fundación Universitaria del Área Andina.

Una de las funciones sustanciales de la educación superior es la extensión universitaria, para lo cual el centro de proyección social se apoyo en al unidad de sistemas e informática con el fin de sistematizar la información de egresados.

1.2 TÍTULO

Los proyectos durante la pasantía que se desarrollaron se denominaron de la siguiente manera:

- Egresados Web
 - Es llamado así, debido a que fue desarrollado e implementado en el Hotel Tequendama durante el IV Encuentro de Egresados, mediante una red LAN, y su objetivo era actualizar la información de los egresados de la Fundación Universitaria del Área Andina.

- Gestión Pública Hoja de Vida de los Egresados
 - Nombrado de esta forma, porque contiene la información completa de un egresado, consolidando los campos más importantes y necesarios de los sistemas de información que se integran, como lo son SINU (Sistema de Información de la Fundación Universitaria del Área Andina), SNIES (Sistema Nacional de Información de la Educación Superior), SPADIES (Sistema de Prevención y Análisis de la Deserción en Las Instituciones de Educación Superior) y demás IES (Instituciones de Educación Superior) y CVLAC (Currículum Vitae Latinoamericano y el Caribe).

1.3 PLANTEAMIENTO DEL PROBLEMA

En la Fundación Universitaria del Área Andina se presentan dificultades en el almacenamiento, modificación y actualización de los datos de un egresado, ya que no existe una herramienta que permita que el egresado rectifique y actualice sus datos; por lo tanto se presentan dos proyectos durante la pasantía, en los cuales se maneja la figura del tercero, como egresado, en los cuales se soluciona este problema de la diversidad de bases de datos con información de los egresados, condensando todo en una única base de datos, que será actualizada y poblada por los mismos, siguiendo unas reglas de negocio adecuadas.

1.4 DESCRIPCIÓN DEL PROBLEMA

Durante las pasantías en la Fundación Universitaria del Área Andina, se presentaron varios problemas, como se menciona a continuación, con respecto a la información de los egresados, a los cuales se le dieron una correcta y adecuada solución teniendo en cuenta los recursos que poseía la Fundación.

El primer problema que presentó el Centro de Proyección Social y Egresados, consistió en formar una base de datos de los Egresados de la Fundación; por lo tanto se le dio una solución que fue migrar un archivo plano en Excel que poseía el Centro, a una base de datos en FireBird.

Luego se presentó el IV Encuentro de Egresados de la Fundación Universitaria del Área Andina realizado en el Hotel Tequendama, para el cual se desarrolló e implementó un software con la colaboración de la División de informática Educativa, denominado Egresados Web, que permitió actualizar la información de los egresados en la base de datos, adicionalmente se aplicó una encuesta, la cual se interpretó estadísticamente, con el objetivo de crear nuevos postgrados y especializaciones en la Fundación; en este encuentro se expuso el proyecto del Portal de Egresados de la Fundación.

Dada la importancia de la figura del egresado como parte de la comunidad universitaria, se ratificó la necesidad de desarrollar un sistema de información con acceso Web que permitiese, de manera centralizada y consolidada en el Sistema de Información SINU, la actualización y modificación de los datos de un egresado contenidos allí, con el fin de obtener mediante consultas la información para las diferentes entidades del estado como el MEN (Ministerio de Educación Nacional) con sus sistemas SNIES (Sistema Nacional de Información de la Educación Superior), SPADIES (Sistema de Prevención y Análisis de la Deserción en Las Instituciones de Educación Superior) y demás IES (Instituciones de educación Superior), así como las internas de la Fundación.

1.5 JUSTIFICACIÓN

Los proyectos se realizan con el fin de colaborar al Centro de Proyección Social y Egresados de la Fundación Universitaria del Área Andina durante el período de la pasantía. Egresados Web, el primer software se crea debido a la necesidad de llevar un control y actualización sobre los datos de los egresados de la Fundación, debido a que en el IV Encuentro de Egresados se manejó un gran e importante flujo de egresados y fue el mejor momento para actualizar estos datos.

Gestión Pública Hoja de Vida de Egresados se crea debido a la necesidad de tener un mejor control de los datos de un egresado, consolidando en una herramienta de información que agrupa todos los datos de los egresados en un solo sistema, permitiendo la actualización de esta información, siendo de fácil manejo.

1.6 OBJETIVOS

1.6.1 OBJETIVO GENERAL

- Apoyar al Centro de Proyección Social y Egresados de la Fundación Universitaria del Área Andina en los diferentes proyectos, generando soluciones informáticas adecuadas.

1.6.2 OBJETIVOS ESPECÍFICOS

- Desarrollar un software que actualice la información de los egresados en el IV Encuentro de Egresados de la Fundación Universitaria del Área Andina.
- Generar una pieza de software que consolide los datos de los egresados, para que éstos puedan actualizarlos vía web.
- Afianzar los conocimientos teóricos obtenidos, con su aplicación en los distintos proyectos de la Fundación Universitaria del Área Andina.

1.7 METODOLOGÍA DE INVESTIGACIÓN

Este trabajo es el resultado de la investigación que se ha llevado a cabo para encontrar una posible solución a la necesidad de unificar los datos de un egresado en la Fundación Universitaria Del Área Andina, por medio del desarrollo de un sistema de información, donde se busca darle un enfoque actual de los Desarrollos de Sistemas.

La metodología de investigación aplicada fue Descriptiva – Analítica, ya que se hizo un análisis e investigación acerca de los métodos adecuados para llegar a cumplir adecuadamente los objetivos propuestos, para plantear y aplicar la mejor solución para el desarrollo de los proyectos.

Con esta metodología se busca saber cual es la mejor forma de identificar los requerimientos para el desarrollo del sistema de información, y posteriormente desarrollar y generar así un sistema de información confiable y eficaz que cumpla con las necesidades del usuario.

1.8 LÍNEA DE INVESTIGACIÓN

En la Corporación Universitaria Minuto de Dios existen cuatro líneas de investigación para la presentación de proyectos, estas cuatro líneas se aplican para todo los programas de la universidad Minuto de Dios. ¹Dentro de estas cuatro líneas se encuentran sublíneas de investigación las cuales son específicas para cada programa, las cuales son:

- Línea 1 - Innovación educativa y transformación social.
- Línea 2 - Lenguaje, comunicación y pensamiento.
- Línea 3 - **Innovaciones tecnológicas y cambio social.**
- Línea 4 - Gestión, participación y desarrollo comunitario.

Para el programa de Tecnología en informática que se encuentra dentro de la Línea 3 - Innovaciones tecnológicas y cambio social, existen cuatro sublíneas las cuales son:

- Desarrollo de software
- **Sistemas de información**
- Redes de computadores
- Plataformas

La línea de investigación que se adapta al proyecto es sistema de información ya que las finalidades de los sistemas de información, como las de cualquier otro sistema dentro de una organización, son procesar entradas, mantener archivos de datos relacionados con la organización y producir información, reportes y otras salidas. Los sistemas estratégicos de información a menudo originan cambios en la organización como los productos, servicios y procedimientos internos, llevándola a nuevos patrones de comportamiento. Las comunidades y organizaciones pueden requerir de un cambio en sus operaciones internas para sacar ventaja de las nuevas tecnologías aplicadas a los sistemas de información. Tales cambios con frecuencia requieren que el recurso humano este constante realizando investigaciones para garantizar un mejor uso de la tecnología, generando nuevas ideas para garantizar la permanencia de la organización en un mercado altamente competitivo y logre una relación mucho más estrecha con los clientes y proveedores.

¹http://academia.uniminuto.edu/index.php?option=com_docman&task=cat_view&gid=81&Itemid=74

Los sistemas de información computarizados toman gran importancia en las organizaciones porque a través de ellos se procesan gran cantidad de datos que sirven para la toma de decisiones frente a un mercado dinámico y cambiante; por ello se requiere que los tecnólogos en informática adquieran la habilidad de investigar para estar constantemente planteando como garantizar nuevas estructuras de sistemas de información que permitan movilidad y dinamismo en las organizaciones junto con estructuras de hardware y software que garanticen la proyección y permanencia de las organizaciones.

2. MARCO REFERENCIAL

2.1 ESTADO DEL ARTE

En el campo académico el egresado es parte fundamental de las Instituciones de Educación. Partiendo de este hecho las Instituciones han desarrollado una serie de piezas de software, con el fin de facilitar la generación de reportes para el Estado. Entre estos tenemos:

²QUADRANTE

El programa de egresados de la Facultad de Ciencias de la Universidad Nacional de Colombia, Sede Bogotá es una estrategia creada en 1999 con el objetivo de generar vínculos productivos de la comunidad académica con los egresados, que aporten a la construcción del devenir científico, tecnológico y económico de la nación.

A. COORDINACIÓN DEL CENTRO DE INFORMACIÓN DE EGRESADOS Y DOCENTES DE LA FACULTAD “QUADRANTE” (Software en fase de prueba).

Confiabilidad y agilidad en la disponibilidad oportuna de información sobre la trayectoria académica y profesional de egresados y docentes de la Facultad, con la debida seguridad (confidencialidad y clave para cada egresado o docente).

B. INFORMACIÓN LABORAL.

A través de la red de egresados se divulgan las ofertas laborales remitidas por instituciones públicas y privadas, previa verificación de información de referencia y establecimiento del perfil requerido.

C. BOLETÍN Y SISTEMA DE NOTICIAS PARA EGRESADOS.³

Boletín periódico en formato electrónico. Noticias divulgadas a través de correo electrónico: Información de interés académico, Servicio de Reseña de Prensa (aspectos sobre Ciencia- Tecnología- Salud en El Tiempo, El Espectador, Un

² <http://www.grupoquadrante.com/>

³ <http://www.directoriociencias.unal.edu.co/programaegresados.php>

Periódico, Un Programa, Hipótesis- Programa Radial-). Información de interés general.

D. APOYO EN ALIANZAS ESTRATÉGICAS DE LA FACULTAD CON INSTITUCIONES PROFESIONALES CON AMPLIA PARTICIPACIÓN DE EGRESADOS Y OTRAS QUE GENEREN BENEFICIOS PARA ELLOS.

E. APOORTE EN LA PERMANENTE CONSTRUCCIÓN DE LA POLÍTICA DE EGRESADOS DEL ALMA MATER.

Interlocución a nivel de Sede y Nacional en pos de la formalización de la Política y ejecución.

F. CARNETIZACIÓN DEL EGRESADO.

G. ATENCIÓN GENERAL AL USUARIO: EGRESADOS, COMUNIDAD ACADÉMICA, INSTITUCIONES EXTERNAS.

*SINU*⁴

Es la herramienta de apoyo a la gestión de custodia y certificación de la información generada en las diferentes áreas como resultado de los procesos académicos y administrativos de las instituciones de educación superior.

Concebido para apoyar los requerimientos del proceso de ACREDITACION Y AUTO EVALUACION INSTITUCIONAL.

Objetivo

Ser la herramienta de apoyo y control de los procesos académicos y administrativos de las entidades de educación superior.

⁴ <http://aciescon.com/default.php?id=400>

Alcance⁵

El sistema SINU esta diseñado para atender las necesidades de administración de información de las áreas académicas y administrativas de las universidades.

Apoya las funciones de la institución en lo relacionado con el manejo de la información de aspirantes, alumnos, egresados, docentes, recursos físicos.

Sirve de control a la función administración llevando un registro detallado de las actividades financieras, contables, presupuestales y de manejo de recursos físicos que se desprenden de la actividad académica.

⁵ <http://aciescon.com/default.php?id=400>

2.2 FUNDAMENTOS TEÓRICOS

*FIREBIRD*⁶

Es una base de datos relacional que ofrece muchas características de SQL ANSI estándar y que funciona en Linux, Windows, MacOSX y una variedad de plataformas UNIX. Firebird ofrece una concurrencia excelente, alto rendimiento y un poderoso lenguaje de procedimientos almacenados y disparadores. Ha estado usándose en producción bajo varios nombres desde 1981.

*ORACLE*⁷

Es un sistema de gestión de base de datos relacional (o RDBMS por el acrónimo en inglés de Relational Data Base Management System), fabricado por Oracle Corporation.

Se considera a Oracle como uno de los sistemas de bases de datos más completos, destacando su:

- * Soporte de transacciones.
- * Estabilidad.
- * Escalabilidad.
- * Es multiplataforma.

Su mayor defecto es su enorme precio, que es de varios miles de euros (según versiones y licencias). Otro aspecto que ha sido criticado por algunos especialistas es la seguridad de la plataforma, y las políticas de suministro de parches de seguridad, modificadas a comienzos de 2005 y que incrementan el nivel de exposición de los usuarios. En los parches de actualización provistos durante el primer semestre de 2005 fueron corregidas 22 vulnerabilidades públicamente conocidas, algunas de ellas con una antigüedad de más de 2 años.

Aunque su dominio en el mercado de servidores empresariales ha sido casi total hasta hace poco, recientemente sufre la competencia del Microsoft SQL Server de Microsoft y de la oferta de otros RDBMS con licencia libre como PostgreSQL,

⁶ <http://www.firebird.com.mx/modules/news/>

⁷ <http://es.wikipedia.org/w/index.php?title=Oracle&oldid=9425723>

MySQL o Firebird. Las últimas versiones de Oracle han sido certificadas para poder trabajar bajo Linux.

*JAVA*⁸

Es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 1990. Las aplicaciones Java están típicamente compiladas en un bytecode, aunque la compilación en código máquina nativo también es posible. En el tiempo de ejecución, el bytecode es normalmente interpretado o compilado a código nativo para la ejecución, aunque la ejecución directa por hardware del bytecode por un procesador Java también es posible.

El lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel como punteros. JavaScript, un lenguaje interpretado, comparte un nombre similar y una sintaxis similar, pero no está directamente relacionado con Java.

Sun Microsystems proporciona una implementación GNU General Public License de un compilador Java y una máquina virtual Java, conforme a las especificaciones del Java Community Process, aunque la biblioteca de clases que se requiere para ejecutar los programas Java no es software libre.

*JSP (JAVA SERVER PAGES)*⁹

Es una tecnología Java que permite a los programadores generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo. Las JSP's permiten al código Java y a algunas acciones predefinidas ser incrustadas en el contenido estático del documento web.

En las JSP se escribe el texto que va a ser devuelto en la salida (normalmente, código HTML) incluyendo código java dentro de él, para poder modificar o generar contenido dinámicamente. El código java se incluye dentro de las marcas de etiqueta `<% y %>`; a esto se le denomina scriptlet.

⁸ <http://es.wikipedia.org/w/index.php?title=Especial:Cite&page=Java&id=9294118>

⁹ www.desarrolloweb.com/articulos/831.php

AJAX (Asynchronous JavaScript And XML)¹⁰

Es una técnica de desarrollo web para crear aplicaciones interactivas. Éstas se ejecutan en el cliente, es decir, en el navegador de los usuarios, y mantiene comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre la misma página sin necesidad de recargarla. Esto significa aumentar la interactividad, velocidad y usabilidad en la misma.

AJAX es una combinación de tres tecnologías ya existentes:

- XHTML (o HTML) y hojas de estilos en cascada (CSS) para el diseño que acompaña a la información.
- Document Object Model (DOM) accedido con un lenguaje de scripting por parte del usuario, especialmente implementaciones ECMAScript como JavaScript y JScript, para mostrar e interactuar dinámicamente con la información presentada.
- El objeto XMLHttpRequest para intercambiar datos asincrónicamente con el servidor web. En algunos frameworks y en algunas situaciones concretas, se usa un objeto iframe en lugar del XMLHttpRequest para realizar dichos intercambios.
- XML es el formato usado comúnmente para la transferencia de vuelta al servidor, aunque cualquier formato puede funcionar, incluyendo HTML preformateado, texto plano, JSON y hasta EBML.

ZKOSS¹¹

Es un framework de aplicaciones web en AJAX, completamente en Java de Código abierto que permite una rica interfaz de usuario para aplicaciones web sin usar JavaScript y con poca programación.

El núcleo de ZK es un mecanismo conducido por eventos basado en AJAX, basado sobre 70 componentes XUL y 80 componentes XHTML, y un lenguaje de marcación para diseñar interfaces de usuario. Los programadores diseñan las páginas de su aplicación en componentes XUL/XHTML ricos en características, y los manipulan con eventos disparados por la actividad del usuario final. Es similar al modelo de programación encontrado en las aplicaciones basadas en GUI de escritorio.

¹⁰ <http://es.wikipedia.org/wiki/AJAX>

¹¹ <http://www.zkoss.org/>

*HIBERNATE*¹²

Es una herramienta de Mapeo objeto-relacional para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) que permiten establecer estas relaciones.

Hibernate es software libre, distribuido bajo los términos de la LGPL (Licencia Pública General Menor de GNU).

¹² <http://www.hibernate.org/5.html>

2.3 REFERENCIA ORGANIZACIONAL

2.3.1 ANTECEDENTES

El Centro de Proyección Social y Egresados de la Fundación Universitaria del Área Andina poseía una serie de hojas de cálculo en Excel, las cuales tenían la información básica de los Egresados de la Fundación Universitaria del Área Andina; esto dio pie a realizar un proyecto en el cual se generara un sistema de información para esta división.

2.3.2 MISIÓN¹³

La Fundación Universitaria del Área Andina es una institución de educación superior comprometida con la formación integral y permanente de personas, con visión emprendedora, humanista y social, mediante la utilización adecuada de la técnica, la tecnología y la ciencia; para dar respuesta a las necesidades del país y la sociedad desde una perspectiva creativa e investigativa.

2.3.3 VISIÓN¹⁴

La Fundación se propone:

El desarrollo de una concepción humanista del hombre, independientemente de la ocupación, disciplina o profesión objeto de su formación.

El diseño de procesos educativos presenciales, semipresenciales y a distancia, acordes con las necesidades socio-económicas de las comunidades y los avances científicos y tecnológicos en todo lo relacionado con los procesos de información, comunicación y educación.

¹³ <http://www.areandina.edu.co>

¹⁴ <http://www.areandina.edu.co>

2.3.4 ESTRUCTURA ORGANIZACIONAL

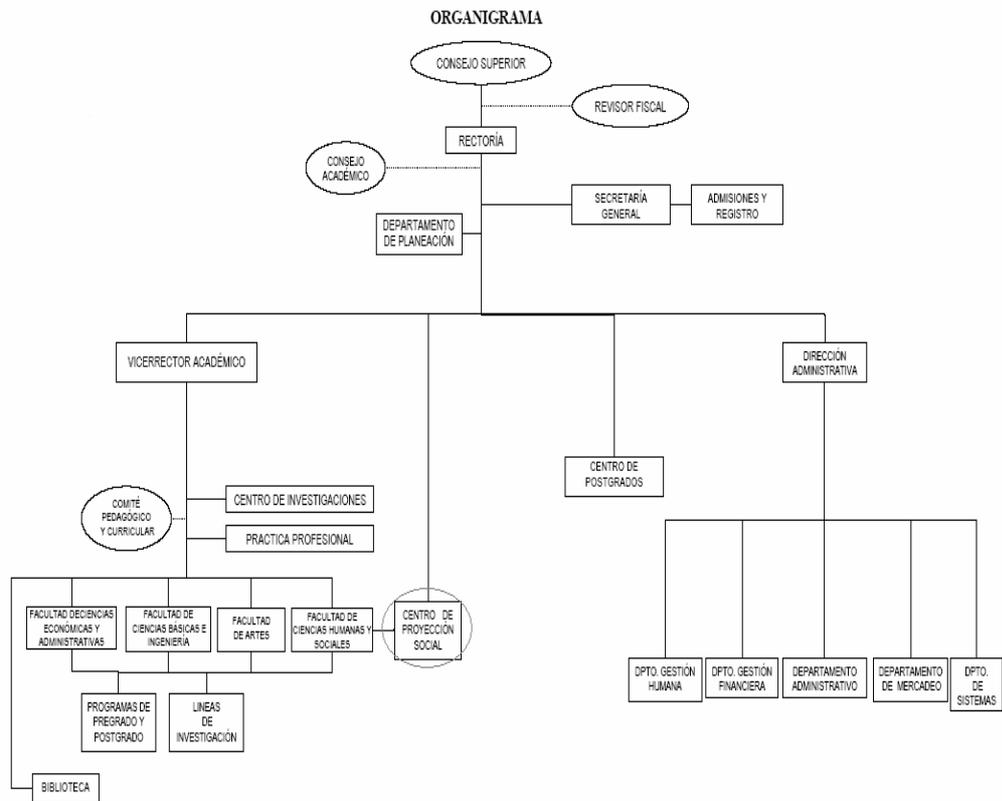


Diagrama No. 1

3. INGENIERÍA DEL PROYECTO

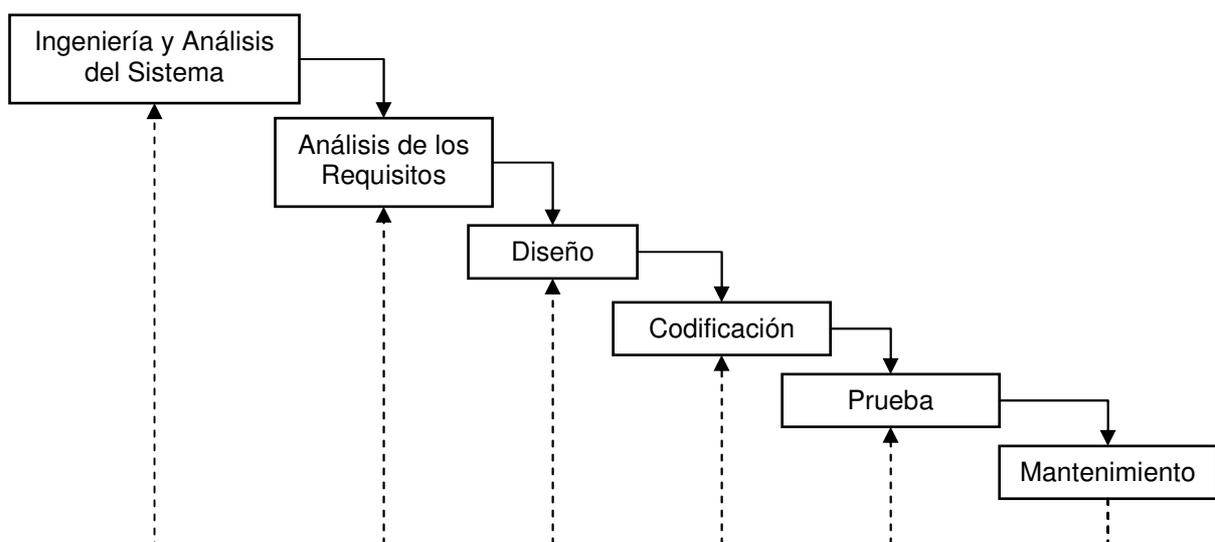
3.1 METODOLOGÍA Y MODELO DE DESARROLLO

Este modelo se aplica al proyecto, Teniendo en cuenta estas cuatro fases se logrará que la plataforma tenga un mayor rendimiento del sistema.

El análisis es muy importante ya que permitirá aclarar dudas acerca del funcionamiento del sistema, generando una idea clara de como puede interactuar las diferentes entidades del sistema y como éstas pueden llegar a ser herramientas fundamentales para obtener resultados óptimos alcanzando finalmente la satisfacción en un usuario final.

Modelo en Cascada¹⁵

El más conocido, esta basado en el ciclo convencional de una ingeniería, el paradigma del ciclo de vida abarca las siguientes actividades:



¹⁵ Ingeniería del Software: Un enfoque práctico, Roger S. Presuman, 3^{ra} Edición, Pág. 26-30.

Diagrama No. 2

*Ingeniería y Análisis del Sistema*¹⁶: Debido a que el software es siempre parte de un sistema mayor el trabajo comienza estableciendo los requisitos de todos los elementos del sistema y luego asignando algún subconjunto de estos requisitos al software.

*Análisis de los requisitos del software*¹⁷: el proceso de recopilación de los requisitos se centra e intensifica especialmente en el software. El ingeniero de software (Analistas) debe comprender el ámbito de la información del software, así como la función, el rendimiento y las interfaces requeridas.

Diseño: el diseño del software se enfoca en cuatro atributos distintos del programa: la estructura de los datos, la arquitectura del software, el detalle procedimental y la caracterización de la interfaz. El proceso de diseño traduce los requisitos en una representación del software con la calidad requerida antes de que comience la codificación.

Codificación: el diseño debe traducirse en una forma legible para la maquina. El paso de codificación realiza esta tarea. Si el diseño se realiza de una manera detallada la codificación puede realizarse mecánicamente.

Prueba: una vez que se ha generado el código comienza la prueba del programa. La prueba se centra en la lógica interna del software, y en las funciones externas, realizando pruebas que aseguren que la entrada definida produce los resultados que realmente se requieren.

Mantenimiento: el software sufrirá cambios después de que se entrega al cliente. Los cambios ocurrirán debido a que hayan encontrado errores, a que el software deba adaptarse a cambios del entorno externo (sistema operativo o dispositivos periféricos), o debido a que el cliente requiera ampliaciones funcionales o del rendimiento.

¹⁶ Ingeniería del Software: Un enfoque practico, Roger S. Presuman, 3^{ra} Edición, Pág. 26-30

¹⁷ Ingeniería del Software: Un enfoque practico, Roger S. Presuman, 1^{ra} Edición, Pág. 33-35

4. ANÁLISIS

4.1 DEFINICIÓN DEL SISTEMA ACTUAL.

El primer proyecto que se realizó, Egresados Web, actualiza y/o crea en su defecto a un egresado, si este no se encuentra en la base de datos de egresados. Adicionalmente, cuenta con una encuesta aplicada en el IV Encuentro de Egresados de La Fundación Universitaria del Área Andina. Este software se encuentra implementado en el Centro de Proyección Social y Egresados.

El segundo proyecto, Gestión Pública Hoja de Vida de Egresados de la Fundación, se integra a SINU, creando un módulo web, con el objetivo de mantener la relación con el egresado y poder actualizar sus datos, para no perder estos contactos, ya que son de vital importancia para la Fundación.

4.2 DESCRIPCIÓN DETALLADA DEL SISTEMA ACTUAL

Egresados Web, se desarrollo en Java, mediante JSP (Java Server Pages), unido a una base de datos en FireBird. Esta pieza de software ofrece una ventana de ingreso por medio del documento de identificación del egresado. Permite visualizar y actualizar la información del egresado, y a continuación nos muestra una encuesta.

Gestión Pública Hoja de Vida de Egresados Web, se integra a SINU, debido a que este nos ofrece únicamente un módulo especial para los egresados, el cual contiene las siguientes funciones:

Parámetros del modulo

- Verificación de terminación de estudios
- Solicitud / Aprobación / Rechazo de grado
- Actas de grado
- Registro de titulo
- Actualización hoja de vida egresados
- Impresión de actas y registro de titulo
- Generación de archivo plano
- Grados especiales
- Certificados
- Reportes

Pero actualmente no posee un módulo Web, que me permita actualizar la información de los egresados, por lo tanto se desarrolla Gestión Pública Hoja de Vida de Egresados de la Fundación Universitaria del Área Andina, con el fin de complementar SINU.

4.3 DESCRIPCIÓN DETALLADA DE CADA PROCESO

El sistema de información Egresados Web necesita capturar el número de documento de un Egresado, una vez capturado ingresa a la información personal de dicho egresado, donde el egresado podrá modificar sus datos personales, permitiendo así actualizar los datos de un egresado.

El sistema de información Gestión Pública Hoja de Vida de Egresados contiene los siguientes procesos para su ingreso:

- **Información no encontrada en las listas de valores**

El egresado podrá agregar o adicionar la información deseada en los siguientes campos:

INFORMACIÓN PERSONAL

- Lugar de Expedición (Cédula)
- Lugar de Nacimiento
- Nacionalidad
- Dependencia Económica
- Lugar Residencia
- Barrio
- Distrito
- Nombre EPS
- Salarios Mínimos
- Discapacidades

ESTUDIOS

- Institución

UNIVERSITARIOS

- Institución
- Profesión
- Nivel de Formación

ESTUDIOS COMPLEMENTARIOS

- Institución

LABORAL

- Entidad
- Niveles de Cargo
- Tipos de Vinculación

EXPERIENCIA DOCENTE

- Niveles de Cargo
- Institución
- Tipo de Vinculación

Esta información será almacenada en una tabla temporal, para luego ser validada por un tercero vinculado con la Fundación Universitaria del Área Andina, que deberá verificar los datos, una vez validados y aceptados los datos que ingresó el egresado serán trasladados a la base de datos SINU.

- **Ingreso a Gestión Pública Hoja de Vida de Egresados**

El egresado que desee ingresar deberá ser validado por el sistema SINU, mediante su usuario y contraseña del correo institucional, los cuales le fueron entregados al inicio del programa académico, en el departamento o programa respectivo, en un sobre, de acuerdo a la normatividad vigente; para luego ser validado contra LDAP. Si el sistema no reconoce al egresado no podrá acceder al sistema.

- **Actualización de los Datos de la Gestión Pública Hoja de Vida de Egresados**

Una vez validado el egresado y si se permitió su ingreso, éste podrá ver sus datos y modificarlos, mediante cuatro formularios:

- Datos Personales
- Estudios
- Información Laboral
- Idiomas

Estos datos quedan almacenados en una base de datos temporal. Gestión Pública Hoja de Vida de Egresados, le mostrará un mensaje al egresado, en el cual se avisa que los datos actualizados por este, no se actualizarán en línea, sino que transcurrirá un tiempo mientras estos son validados.

- **Validación**

El proceso de validación consta de la confirmación y actualización de los datos del egresado. Se tienen dos roles en cuanto a la seguridad de Gestión Pública Hoja de Vida de Egresados, el primero, el cual modificará los datos de Información Laboral, si el egresado se encuentra actualmente vinculado a la universidad; el segundo, actualizará los demás egresados. Para la validación, la persona encargada, ingresará con su usuario y contraseña, y observará un listado de las últimas actualizaciones de datos realizadas por el egresado en Gestión Pública Hoja de Vida del Egresado, para su respectiva rectificación; el validador podrá ver los datos completos del egresado mediante un formulario, el cual señalará los últimos campos modificados por el egresado, para facilitar la función del validador, quien aprobará o no esta información. Después de ser revisada la información y modificada correctamente, según el estándar definido por la universidad, ésta será almacenada en SINU, en caso de ser correcta; en caso contrario, se le notificará al Egresado sobre la información inconsistente.

- **Visualizar la Información del Egresado**

Después de que el validador confirme los datos actualizados en la Base de Datos Temporal, y migre estos datos a SINU, el egresado podrá visualizar sus datos actualizados.

4.4 DIAGRAMA DE ENTRADA SALIDA

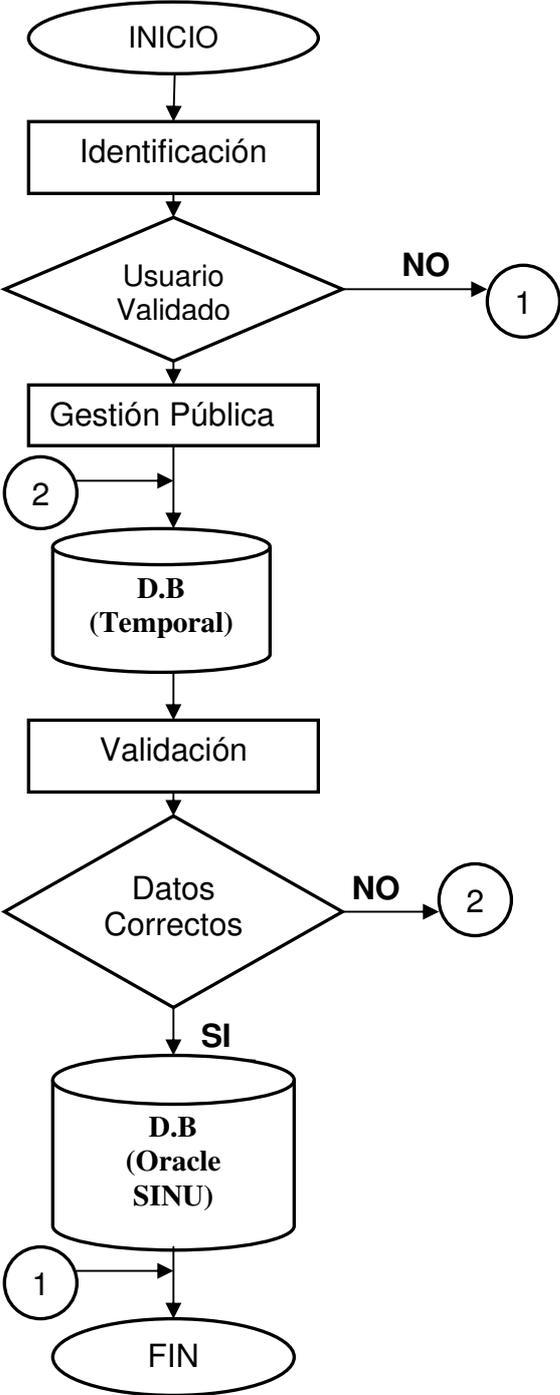


Diagrama No. 3

5. DISEÑO

5.1 Diccionario de datos

A continuación se muestra el conjunto de metadatos que contiene la base de datos mostrando las diferentes características lógicas de los datos que se van a utilizar en el sistema que se desarrolló.

Tabla No. 1

EGRESADO		
PK	IDGRESADO	Llave Primaria, me representa el número del Egresado
*	NUMDOCUMENTO	Atributo Obligatorio, el cual es la cédula del egresado
*	PASSWORD	Atributo Obligatorio, el cual me permite tener una contraseña por seguridad en el sistema
*	NOMBRE1	Atributo Obligatorio, todo egresado tiene mínimo un nombre
O	NOMBRE2	Atributo Opcional, No todos los egresados tienen un segundo nombre
*	APELLIDO1	Atributo Obligatorio, todo egresado tiene mínimo un apellido
*	APELLIDO2	Atributo Obligatorio, todo egresado tienen mínimo un segundo apellido
*	TEL_FIJO	Atributo Obligatorio, todo egresado debe tener un número fijo de su residencia
O	CELULAR1	Atributo Opcional, No todos los egresados tienen celular
O	CELULAR2	Atributo Opcional, No todos los egresados tienen un segundo celular
O	FAX	Atributo Opcional, No todos los egresados tienen Fax
*	DIRECCION	Atributo Obligatorio, Todo egresado tiene una dirección de residencia
*	FECHA_NACIM	Atributo Obligatorio, Todo egresado tiene una fecha de nacimiento
*	E_MAIL	Atributo Obligatorio, Todo egresado debe tener una cuenta de correo electrónico
*	FECHA_GRADO	Atributo Obligatorio, Todo egresado tiene una fecha de grado, sino no sería egresado

PROGRAMA		
PK	IDPROGRAMA	Llave Primaria, me representa el número o código del programa
*	NOM_PROGRAMA	Atributo Obligatorio, el cual es el nombre del programa
*	DURACION	Atributo Obligatorio, el cual me da el tiempo de duración del programa
*	MISION	Atributo Obligatorio, el cual me describe la misión del programa
*	VISION	Atributo Obligatorio, el cual me describe la visión del programa
*	OBJETIVOS	Atributo Obligatorio, el cual nombra los diferentes objetivos de un programa
*	PERFIL_EGRESADO	Atributo Obligatorio, contiene el perfil estudiantil de un egresado
*	TELEFONO	Atributo Obligatorio, contiene el número telefónico de cada programa
*	EXT1	Atributo Obligatorio, hace referencia a la extensión telefónica de cada programa
O	EXT2	Atributo Opcional, No todos los programas contienen dos extensiones telefónicas

EMPRESA		
PK	IDEMPRESA	Llave Primaria, la cual representa el código de referencia de una empresa
*	NOMBRE	Atributo Obligatorio, me permite capturar el nombre de la empresa
*	LUGAR	Atributo Obligatorio, me permite capturar la ubicación de la empresa
O	E_MAIL	Atributo Opcional, no todas las empresas tienen e-mail
O	DIRECCION	Atributo Opcional, no todas las empresas desean poner la dirección donde están ubicadas
*	TEL_EMPRESA	Atributo Obligatorio, todo empresa debe tener un teléfono para ser localizada

PROFES		
PK	IDPROFES	Llave Primaria, la cual representa el código asignado a cada profesor
*	NOMPROFE	Atributo Obligatorio, todo profesor

		tiene un nombre
O	TELEFONO	Atributo Opcional, todo profesor puede no tener número telefónico
O	CELULAR	Atributo Opcional, todo profesor puede no tener celular
O	EMAIL	Atributo Opcional, todo profesor puede no tener e-mail

HIS_LABORAL		
PK	IDHISTORIA	Llave Primaria, la cual representa el código de historia laboral del egresado
FK	IDEGREASADO	Llave Foránea, la cual hace referencia a la identificación del egresado
FK	IDEMPRESA	Llave Foránea, la cual hace referencia a la identificación de la empresa en la que estuvo el egresado
FK	IDCARGO	Llave Foránea, la cual hace referencia al código del cargo que desempeña el egresado
*	DURACION	Atributo Obligatorio, hace referencia al tiempo que estuvo el egresado trabajando
*	TEL_DEPTO	Atributo Obligatorio, hace referencia al teléfono de la empresa donde laboró el egresado
O	E_MAIL	Atributo Opcional, hace referencia al e-mail de la empresa

PROYECTO_EGRESADO		
PK	IDPROYECTOEGRESADO	Llave Primaria, la cual hace referencia al código del proyecto del egresado
FK	IDEGRESADO	Llave Foránea, la cual hace referencia a la identificación del egresado
*	NOMBRE	Atributo Obligatorio, el cual contiene el nombre del proyecto
*	IDTIPO	Atributo Obligatorio, el cual hace referencia al tipo de proyecto realizado por el egresado

		(tesis, monografía, etc.)
*	CALIFICACION	Atributo Obligatorio, el cual contiene la calificación del proyecto del egresado
*	FECHA_INICIO	Atributo Obligatorio, el cual contiene la fecha de inicio del proyecto
*	FECHA_FIN	Atributo Obligatorio, el cual contiene la fecha de culminación del proyecto

DECANO		
PK	IDDECANO	Llave primaria, la cual hace referencia al código del decano por facultad
*	NOMBRE	Atributo Obligatorio, el cual hace referencia al nombre del decano
O	TELEFONO	Atributo Opcional, el cual contiene el número telefónico del decano
O	CORREO	Atributo Opcional, el cual contiene el correo del decano
O	CELULAR	Atributo Opcional, el cual contiene el celular del decano

HIS_PROFES		
PK	IDHISPROFES	Llave Primaria, la cual hace referencia al código de la historia de los profesores
FK	IDPROGRAMA	Llave Foránea, la cual hace referencia al programa en el que estuvo el profesor
FK	ID_PROFES	Llave Foránea, la cual hace referencia al código del profesor
*	FECHA_INICIO	Atributo Obligatorio, el cual hace referencia a la fecha de inicio del profesor en el programa
*	FECHA_SALIDA	Atributo Obligatorio, el cual hace referencia a la fecha de salida del profesor en el programa

HISTORIAL_VISITA		
PK	IDVISITA	Llave Primaria, la cual hace referencia al código de visita del egresado
FK	IDEGRESADO	Llave Foránea, la cual me representa el código del egresado
*	FECHA_VISITA	Atributo Obligatorio, el cual me muestra la

		fecha de visita del egresado
*	HORA_VISITA	Atributo Obligatorio, el cual me muestra la
		hora de visita del egresado
*	DURACION	Atributo Obligatorio, el cual me representa
		la duración del egresado en la página

SERVICIOS		
PK	IDSERVICIOS	Llave primaria, la cual hace referencia al código del servicio
*	NOM_SERVICIO	Atributo Obligatorio, el cual es el nombre del servicio
*	COSTO	Atributo Obligatorio, el cual hace referencia al valor o costo monetario que tiene ese servicio
*	CONTACTO	Atributo Obligatorio, hace posible saber quien es el encargado de brindar dicho servicio
*	OFICINA	Atributo Obligatorio, el cual hace referencia al lugar u oficina donde se puede dirigir para recibir mas información sobre los servicios
*	TELEFONO	Atributo Obligatorio, hace referencia al teléfono del lugar donde se hacen dichos servicios
*	EXT	Atributo Obligatorio, hace referencia a la extensión telefónica del sitio donde se prestan estos servicios
*	TIME_PROCES	Atributo Obligatorio, el cual tendrá el tiempo en que tarde la elaboración de un servicio

HIS_DECANO		
PK	IDHISDECANO	Llave primaria, la cual hacer referencia al código de la historia del decano
FK	IDPROGRAMA	Llave Foránea, la cual representa el código del programa
FK	IDDECANO	Llave Foránea, la cual representa el código del decano
*	FECHA_INICIO	Atributo Obligatorio, hace referencia a la fecha de que ingreso de un decano

0	FECHA_SALIDA	Atributo Opcional, hace referencia a la fecha de salida de un decano
---	--------------	--

CARGO		
PK	IDCARGO	Llave Primaria, la cual hace referencia al código del cargo
*	NOMCARGO	Atributo Obligatorio, hace referencia al nombre del cargo el cual ocupa un egresado

TIPO_PROYECTO		
PK	IDTIPO	Llave Primaria, la cual hace referencia al código del tipo de proyecto
*	DETALLE	Atributo Obligatorio

Tabla No. 2

BAS_SRC_GENERICA	
	TIPO
tip_tabla	VARCHAR2(6) NOT NULL,
cod_tabla	VARCHAR2(6) NOT NULL,
nom_tabla	VARCHAR2(60) NOT NULL,
cod_auxiliar1	VARCHAR2(20) NULL,
cod_auxiliar2	VARCHAR2(20) NULL,
nom_alias	VARCHAR2(30) NULL,
fec_ult_act	DATE NULL,
fec_creacion	DATE NULL,
usu_creacion	VARCHAR2(20) NULL,
usu_actualizacion	VARCHAR2(20) NULL

BAS_TEM_TERCERO	
	TIPO
Id_tercero	NUMBER(11,0) NOT NULL,
num_identificacion	VARCHAR2(12) NOT NULL,
Tip_identificacion	VARCHAR2(6) NOT NULL,
nom_largo	VARCHAR2(200) NOT NULL,
sig_tercero	VARCHAR2(10) NULL,
nom_tercero	VARCHAR2(50) NOT NULL,
pri_apellido	VARCHAR2(30) NULL,
seg_apellido	VARCHAR2(30) NULL,
gen_tercero	VARCHAR2(6) NOT NULL,
est_civil	VARCHAR2(6) NULL,
fec_nacimiento	DATE NULL,
cod_nacionalidad	VARCHAR2(6) DEFAULT 57 NULL,
ind_vive	NUMBER(1,0) DEFAULT 1 NULL,

dir_residencia	VARCHAR2(50) NULL,
dir_email	VARCHAR2(100) NULL,
tel_residencia	VARCHAR2(30) NULL,
tel_cecular	VARCHAR2(20) NULL,
Id_ubi_res	NUMBER(11,0) NULL,
Id_ubi_nac	NUMBER(11,0) NULL,
num_tarjeta_militar	VARCHAR2(20) NULL,
dis_tarjeta_militar	VARCHAR2(5) NULL,
cod_anterior	VARCHAR2(12) NULL,
cod_tercero	VARCHAR2(12) NULL,
fec_creacion	DATE NULL,
fec_modificacion	DATE NULL,
usu_creacion	VARCHAR2(20) NULL,
usu_modificacion	VARCHAR2(20) NULL,
nat_tercero	NUMBER(1,0) NULL,
gru_sanguineo	VARCHAR2(2) NULL,
Frh_sanguineo	VARCHAR2(2) NULL,
zon_origen	VARCHAR2(2) NULL,
Id_ubi_documento	NUMBER(11,0) NULL,
dig_chequeo	CHAR(1) NULL,
ind_sordera	NUMBER(1,0) NULL,
ind_prob_motores	NUMBER(1,0) NULL,
ind_invidente	NUMBER(1,0) NULL,
ind_vision_parcial	NUMBER(1,0) NULL,
eps_tercero	VARCHAR2(6) NULL,
num_est_economico	NUMBER(1,0) NULL,
fec_exp_documento	DATE NULL,
fec_ult_act_laboral	DATE NULL,
ind_act_laboral	NUMBER(1,0) NULL,
cod_cat_tercero	VARCHAR2(5) NULL,

BAS_TEM_GEOPOLITCA	
	TIPO
id_geopolitica	NUMBER(11,0) NOT NULL,
cod_pais	NUMBER(3,0) NOT NULL,
cod_departamento	NUMBER(3,0) NOT NULL,

BAS_TEM_INF_ACADEMICA	
	TIPO
id_inf_academica	NUMBER(11,0) NOT NULL,
Id_tercero	NUMBER(11,0) NOT NULL,
Id_entidad	NUMBER(11,0) NOT NULL,

cod_ciudad	NUMBER(3,0) NOT NULL,
nom_div_geopolitica	VARCHAR2(30) NOT NULL,
num_habitantes	NUMBER(11,0) NULL,
tip_division	VARCHAR2(6) NULL,
cod_division	VARCHAR2(6) NULL,
cod_zon_geografica	VARCHAR2(6) NULL,
fec_ult_act	DATE NULL,
fec_creacion	DATE NULL,
usu_creacion	VARCHAR2(20) NULL,
usu_actualizacion	VARCHAR2(20) NULL,
cod_area	VARCHAR2(10) NULL

ano_termino	NUMBER(4,0) NOT NULL,
num_duracion	NUMBER(4,0) NOT NULL,
niv_formacion	NUMBER(1,0) NOT NULL,
cat_formacion	NUMBER(1,0) NOT NULL,
reg_titulo	VARCHAR2(20) NULL,
num_tar_profesional	VARCHAR2(30) NULL,
Tit_obtenido	VARCHAR2(500) NULL,
gra_esc_nacional	VARCHAR2(10) NULL,
fec_creacion	DATE NULL,
fec_ult_act	DATE NULL,
usu_creacion	VARCHAR2(20) NULL,
usu_actualizacion	VARCHAR2(20) NULL,
cod_profesion	VARCHAR2(6) NULL,
fec_termino	DATE NULL,
num_act_grado	VARCHAR2(30) NULL,
num_pun_escalafon	NUMBER(5,2) NULL

BAS_TEM_BARRIO	
	TIPO
id_barrio	NUMBER(6,0) NOT NULL,
cod_ciudad	NUMBER(11,0) NOT NULL,
cod_barrio	VARCHAR2(6) NULL,
des_barrio	VARCHAR2(60) NULL,
fec_ult_act	DATE NULL,
fec_creacion	DATE NULL,
usu_creacion	VARCHAR2(20) NULL,
usu_actualizacion	VARCHAR2(20) NULL,

BAS_TEM_LABORAL	
	TIPO
Id_inf_laboral	NUMBER(11,0) NOT NULL,
id_tercero	NUMBER(11,0) NOT NULL,
id_entidad	NUMBER(11,0) NOT NULL,
nom_dependencia	VARCHAR2(100) NOT NULL,
nom_sede	VARCHAR2(100) NULL,
nom_cargo	VARCHAR2(100) NULL,
niv_cargo	NUMBER(11,0) NULL,
fec_ingreso	DATE NULL,
fec_retiro	DATE NULL,

cod_loc	VARCHAR2(6) NULL
---------	------------------

num_tiempo	NUMBER(4,0) NULL,
tip_vinculacion	VARCHAR2(6) NULL,
cod_vinculacion	VARCHAR2(6) NULL,
val_remuneracion	NUMBER(18,2) NULL,
act_realizadas	LONG NULL,
est_inf_laboral	NUMBER(1,0) NOT NULL,
fec_creacion	DATE NULL,
fec_ult_act	DATE NULL,
usu_creacion	VARCHAR2(20) NULL,
usu_actualizacion	VARCHAR2(20) NULL,
tip_empresa	NUMBER(1,0) NULL,
clas_vinculacion	NUMBER(1,0) NULL,
num_telefono	VARCHAR2(30) NULL,
num_pun_escalafon	NUMBER(5,2) NULL

5.3 CONCEPTUALIZACIÓN DEL SISTEMA PROPUESTO

La solución planteada para registrar la hoja de vida de egresados se fundamenta en hacer uso de herramientas orientadas a la Web, empleando lenguajes de programación orientada a objetos y enlazadas a bases de datos de tipo relacional lo cual permite un mejor manejo de la información permitiendo así un desempeño óptimo del sistema información.

5.4 DISEÑO DEL SISTEMA PROPUESTO

A continuación se muestra las diferentes interfaces del sistema

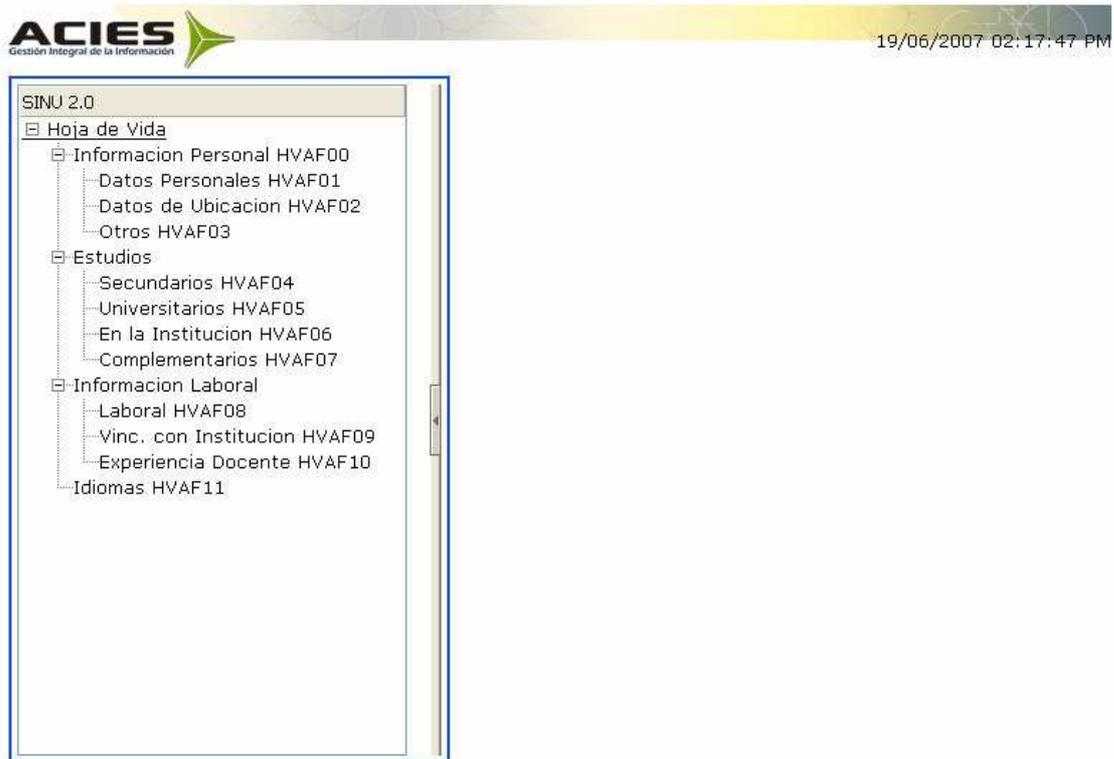


Diagrama No. 5

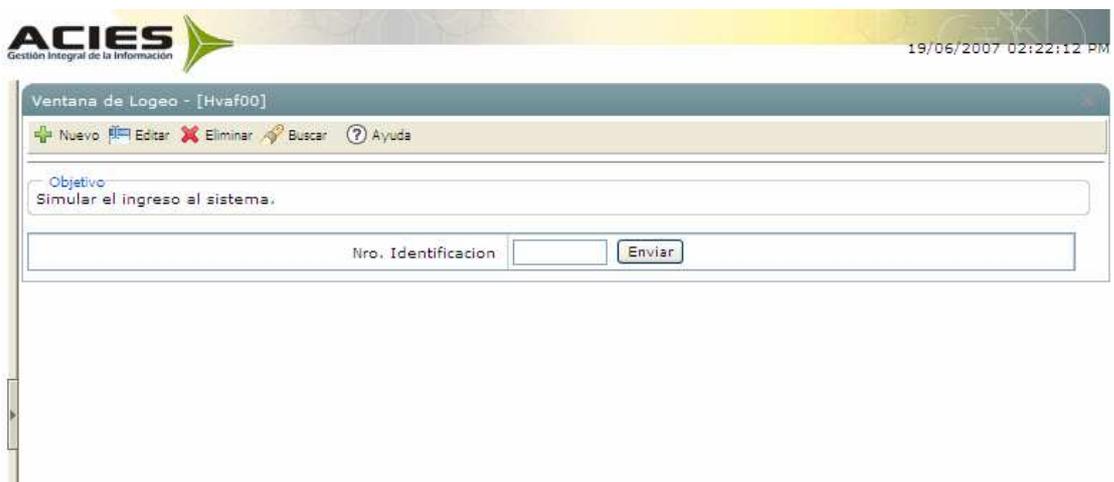


Diagrama No.6

Información Personal - [Hvaf01]

Nuevo
 Editar
 Eliminar
 Buscar
 Ayuda

Objetivo
Recopilar Información Personal de un Egresado.

Nro. Identificación Nombre

Primer apellido	<input type="text"/>	Segundo apellido	<input type="text"/>
Nombre	<input type="text"/>	Tipo Identificación	<input type="text"/>
Lugar de Expedición	<input type="text"/>	Fecha de Nacimiento	<input type="text"/>
Lugar de Nacimiento	<input type="text"/>	Nacionalidad	<input type="text"/>
Género	<input type="text"/>	Estado Civil	<input type="text"/>
Libreta Militar	<input type="text"/>	Distrito	<input type="text"/>

Diagrama No. 7

5.5 DIAGRAMA DE CADA PROCESO

Egresados Web:

Ingreso:

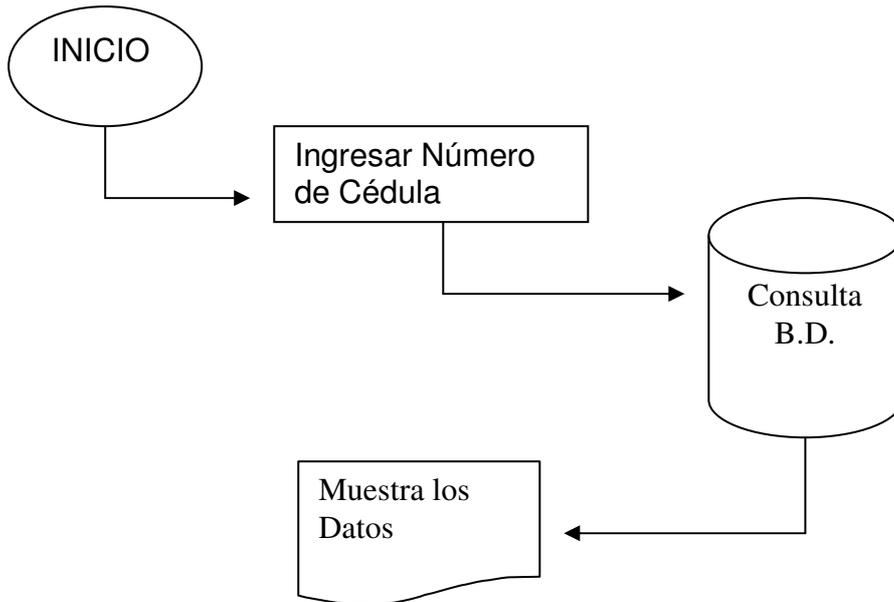


Diagrama No. 8

Modificar/Actualizar:

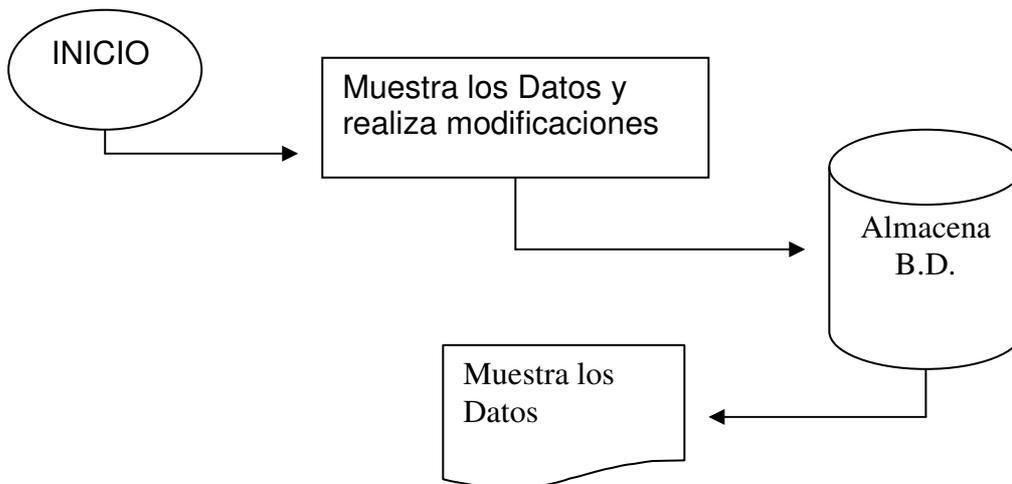


Diagrama No. 9

6. DESARROLLO

6.1 ESPECIFICACIONES TÉCNICAS

6.1.1 SOFTWARE

Para los diferentes desarrollos hechos en la Fundacion Universitaria Del Area Andina se utilizaron los siguientes IDEs como entornos de trabajo:

- NetBeans 5.5
- EasyEclipse Server Java 1.2

Para el manejo y almacenamiento de los datos se utilizaron los siguientes motores gestores de base de datos:

- FireBird 3.2
- Oracle 9i

6.1.2 HARDWARE

El hardware que necesita para que el software funcione correctamente es:

CLIENTE

- 128 MB en RAM
- 2 GB en Disco Duro
- Procesador Pentium III 2.0 Ghz

SERVIDOR

- 256 en RAM
- 10 GB de Disco Duro
- Procesador Pentium IV 3.0 Ghz
- Tarjeta Red 10/100/1000 Mbps

6.2 ESTRUCTURA DEL CÓDIGO FUENTE

Egresados Web esta estructurado de la siguiente manera:

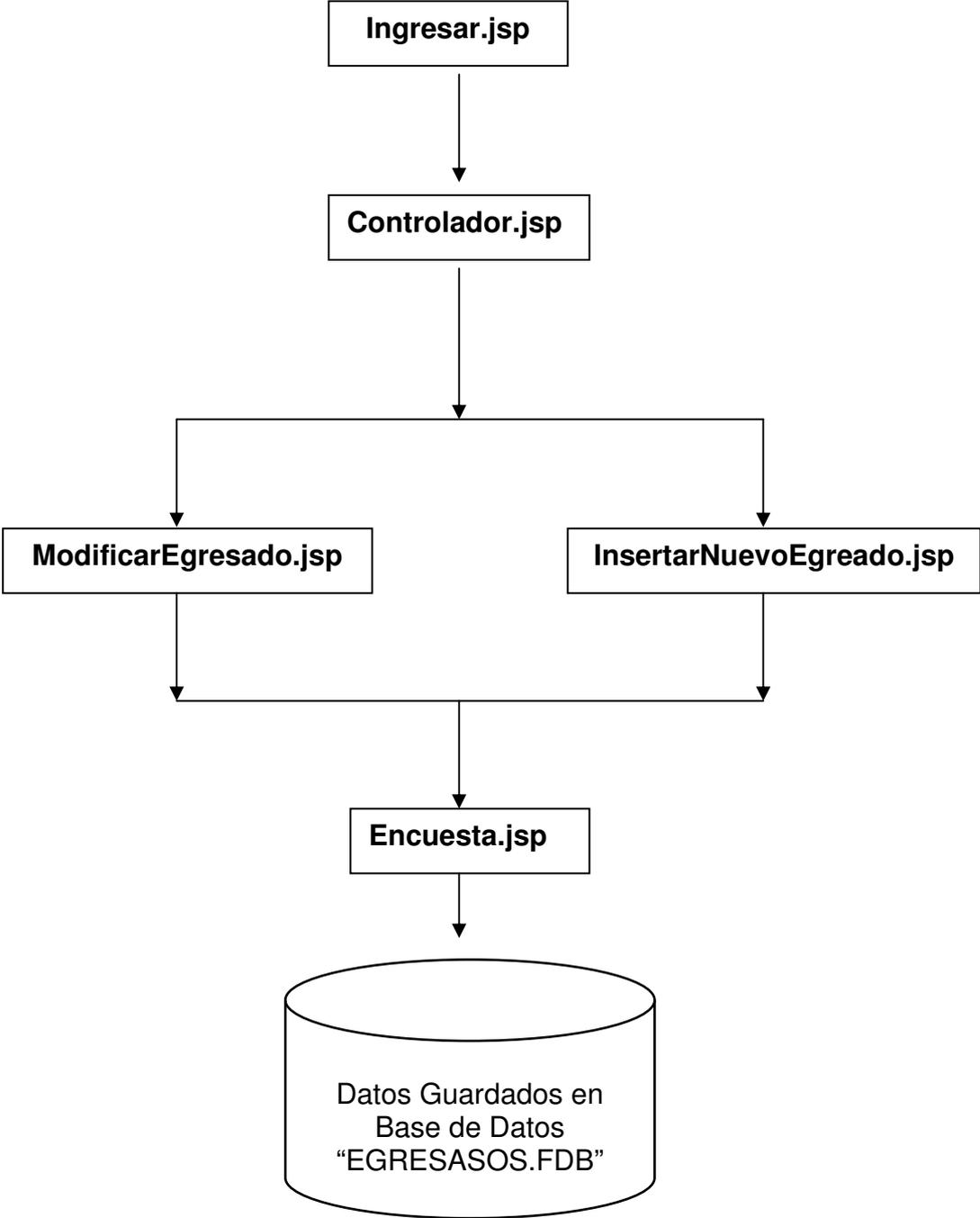


Diagrama No. 10

GESTION PÚBLICA HOJA DE VIDA EGRESADOS esta estructurado de la siguiente manera:

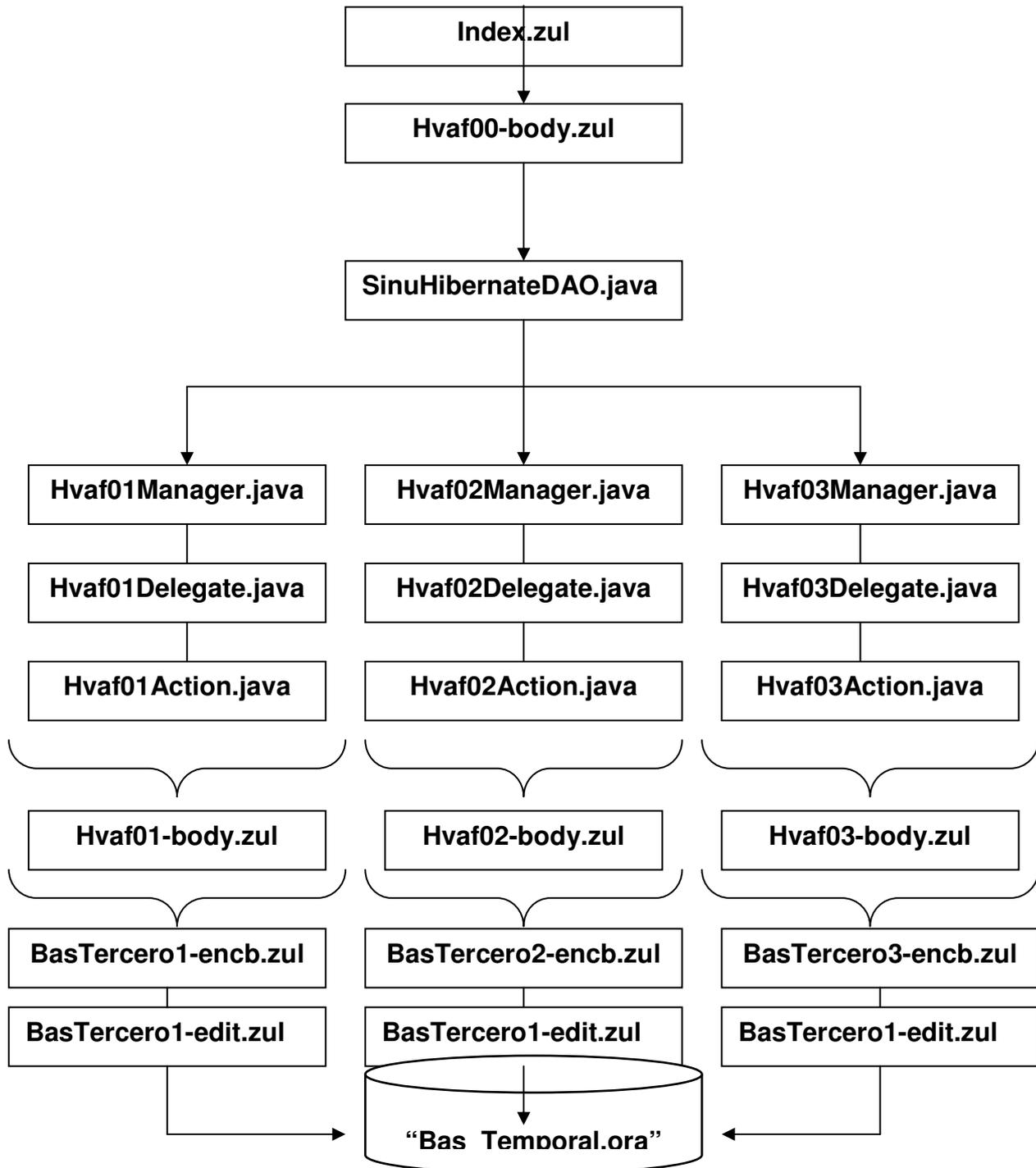


Diagrama No. 11

7. PRUEBAS

La prueba de software es un conjunto de herramientas, técnicas y métodos que permite saber el desempeño y rendimiento de un software, debido a esto en el mes de octubre de 2006, en la semana del 17 al 23 se realizaron pruebas al sistema Egresados Web, por parte de los desarrolladores del sistema y por los usuarios.

Las pruebas consistieron en:

- **Verificación y Validación**

Esta prueba fue hecha por parte de los desarrolladores, en esta prueba se realizó la revisión de los planes, del código, de los requerimientos, de la documentación y las especificaciones. Posteriormente se planeó una reunión con los usuarios para evaluar dichos documentos y requerimientos.

- **Evaluación por parte de usuario**

Esta prueba fue realizada con el propósito de familiarizar a los diferentes usuarios que utilizarían el sistema de información Egresados Web, esto ayudaría a encontrar posibles errores que no fueron identificados por los desarrolladores. Una vez hecho los ajustes al sistema de información se capacitó a los usuarios sobre el manejo del software.

CRONOGRAMA

Cronograma de desarrollo del software Egresados Web

Fecha	Actividad
21/06/06	Análisis y comprensión del problema
22/06/06	Entrega cronograma de actividades
28/06/06	Objetivos del análisis
29/06/06	Entender el uso del sistema
05/07/06	Entender el uso del sistema
06/07/06	Desarrollo de los casos de uso
10/07/06	Desarrollo de los casos de uso
12/07/06	Analizar los cambios de estado de los objetivos
13/07/06	Analizar los cambios de estado de los objetivos
17/07/06	Analizar los cambios de estado de los objetivos
19/07/06	Definir las interacciones entre los objetos
20/07/06	Definir las interacciones entre los objetos
24/07/06	Analizar la integración con los sistemas cooperativos
26/07/06	Analizar la integración con los sistemas cooperativos
27/07/06	Revisión del análisis a entregar
31/07/06	Primera entrega del análisis

Cronograma de desarrollo del software Gestión Publica Hoja de Vida de Egresados

Cronograma de Actividades		
Semana	Fecha	Actividad
1	01 - 09 de Febrero	Comparativo entre SNIES, SINU, CVLAC, EGRESADOS WEB
2	12 - 16 de Febrero	Lectura y Análisis sobre Los Estándares de Programación
3	19 - 23 de Febrero	Lectura y Análisis de blueprints de Java
4	26 - 02 de Marzo	Lectura y Análisis sobre Hibernate
5	05 - 09 de Marzo	Lectura y Análisis de zkoss
6	12 - 30 de Marzo	Instalación y Configuración de Ambientes de Trabajo
7	02 - 13 de Abril	Codificación, Verificación, Ensamble del primer Módulo del sistema
8	16 - 20 de Abril	Codificación, Verificación, Ensamble del primer Módulo del sistema
9	23 - 02 de Mayo	Codificación, Verificación, Ensamble del primer Módulo del sistema
10	04 - 11 de Mayo	Codificación, Verificación, Ensamble del primer Módulo del sistema
11	14 - 18 de Mayo	Codificación, Verificación, Ensamble y envío del primer Módulo del sistema
12	22 - 25 de Mayo	Codificación, Verificación, Ensamble del segundo Módulo
13	28 - 08 de Junio	Codificación, Verificación, Ensamble del segundo Módulo
14	12 - 16 de Junio	Codificación, Verificación, Ensamble del segundo Módulo
15	19 - 22 de Junio	Codificación, Verificación, Ensamble del segundo Módulo
16	25 - 30 de Junio	Codificación, Verificación, Ensamble y envío del Segundo Módulo del sistema

GLOSARIO

AJAX: acrónimo de ***A**ynchronous **J**ava**S**cript **A**nd **X**ML* (JavaScript y XML asíncronos), es una técnica de desarrollo web para crear aplicaciones interactivas.

API: (del inglés Application Programming Interface - Interfaz de Programación de Aplicaciones) es el conjunto de funciones y procedimientos (o métodos si se refiere a programación orientada a objetos) que ofrece cierta librería para ser utilizado por otro software como una capa de abstracción.

Javadoc: Es una utilidad de Sun Microsystems para generar APIs en formato HTML de un documento de código fuente Java. Javadoc es un standard industrial para documentar clases de Java. La mayoría de los IDEs los generan automáticamente.

JDBC: es el acrónimo de Java Database Connectivity, un API que permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java independientemente del sistema de operación donde se ejecute o de la base de datos a la cual se accede utilizando el dialecto SQL del modelo de base de datos que se utilice.

JSP: (JavaServer Pages), es una tecnología Java que permite a los programadores generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo. Las JSP's permiten al código Java y a algunas acciones predefinidas ser incrustadas en el contenido estático del documento web.

JRE: (Java Runtime Environment, o Entorno en Tiempo de Ejecución de Java) es el software necesario para ejecutar cualquier aplicación desarrollada para la plataforma Java. El usuario final usa el JRE como parte de paquetes software o plugins (o conectores) en un navegador Web. Sun ofrece también el SDK de Java 2, o JDK (Java Development Kit) en cuyo seno reside el JRE, e incluye herramientas como el compilador de Java, Javadoc para generar documentación o el depurador.

Script: es un programa usualmente simple, que generalmente se almacena en un archivo de texto plano. Los guiones son casi siempre interpretados, pero no todo programa interpretado es considerado un guión. El uso habitual de los guiones es realizar diversas tareas como combinar componentes, interactuar con el sistema operativo o con el usuario. Por este uso es frecuente que los shells sean a la vez intérpretes de éste tipo de programas.

TOMCAT: (también llamado Jakarta Tomcat o Apache Tomcat) funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las especificaciones de los servlets y de JavaServer Pages (JSP) de Sun Microsystems.

JavaScript: es un lenguaje interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C.

XML: sigla en inglés de eXtensible Markup Language («lenguaje de marcas extensible»), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C). Es una simplificación y adaptación del SGML y permite definir la gramática de lenguajes. Por lo tanto XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades.

CONCLUSIONES

- Se apoyó al Centro de Proyección Social y Egresados de la Fundación Universitaria del Área Andina en los diferentes proyectos, generando soluciones informáticas que ayudaron a mejorar el manejo, almacenamiento y manipulación de información.
- Se desarrolló la pieza de software *Egresados Web*, la cual fue implementada en el IV encuentro de Egresados de la Fundación Universitaria del Área Andina en el Hotel Tequendama.
- La aplicación desarrollada permitió generar la interpretación de los datos registrados en el IV encuentro de Egresados de la Fundación Universitaria del Área Andina, con el fin de crear postgrados en la Fundación Universitaria del Área Andina.
- Se generó una herramienta orientada a la Web compatible con el sistema de información SINU, que posee la Fundación Universitaria del Área Andina, y permitiendo la actualización de los datos de los Egresados de la Fundación Universitaria del Área Andina.
- Se afianzaron los conocimientos tanto teóricos como prácticos en al area de informática.

BIBLIOGRAFÍA

PREESMAN, Roger S. Ingeniería del Software: Un enfoque practico Barcelona:
3^{ra} Edición, Pág. 26-30

CAMPDERRICH, Benet. Ingeniería del Software
Pág. 206-230

COLABORADORES DE WIKIPEDIA, Última revisión: 30 de mayo del 2007, 23:22
UTC
<http://es.wikipedia.org/w/index.php?title=Java&oldid=9294118>

COLABORADORES DE WIKIPEDIA, Última revisión: 18 de mayo del 2007, 19:08
UTC
<http://es.wikipedia.org/w/index.php?title=Java&oldid=9294118>

MANUAL DEL SISTEMA

FUAA FUNDACIÓN UNIVERSITARIA DEL ÁREA ANDINA

Autores:	Jorge Andrés Cañas Vargas Walter Julián Bernal Suárez
Fecha Creación:	04/05/2007
Ultima Actualización:	04/05/2007
Código:	MAN_SYSTEM_WEB_V1.0
Versión:	1.1

MANUAL DEL SISTEMA

El siguiente manual funcional tiene como propósito precisar los alcances operativos de algunos procesos denominado por el cliente como (gestión pública hoja de vida de egresados), dichos procesos consisten en la actualización, control, administración y validación de los datos de un egresado de la Fundación Universitaria del Área Andina, mediante la interacción del sistema SINU y el responsable del tema (El Centro de Proyección Social).

Una de las principales características del sistema (gestión pública hoja de vida de egresados) es permitir a un egresado de la Fundación Universitaria del Área Andina poder actualizar algunos datos vía Web; además generar consultas de los egresados para SPADIES, SNIES, y demás.

SINÚ como sistema de información es el eje y el centro de las actividades académico – administrativas de la institución, por su parte el software a desarrollar (gestión pública hoja de vida de egresados) será un modulo anexo al sistema. Este modulo ayudará a tener actualizados los datos de un egresado y generar los reportes hacia las diferentes unidades, entidades y estatales.

El funcionamiento del sistema se ve en la grafica No.1, el proceso de inserción, validación y control de los datos de un egresado. Este proceso permite asegurar la confiabilidad sobre estos datos.

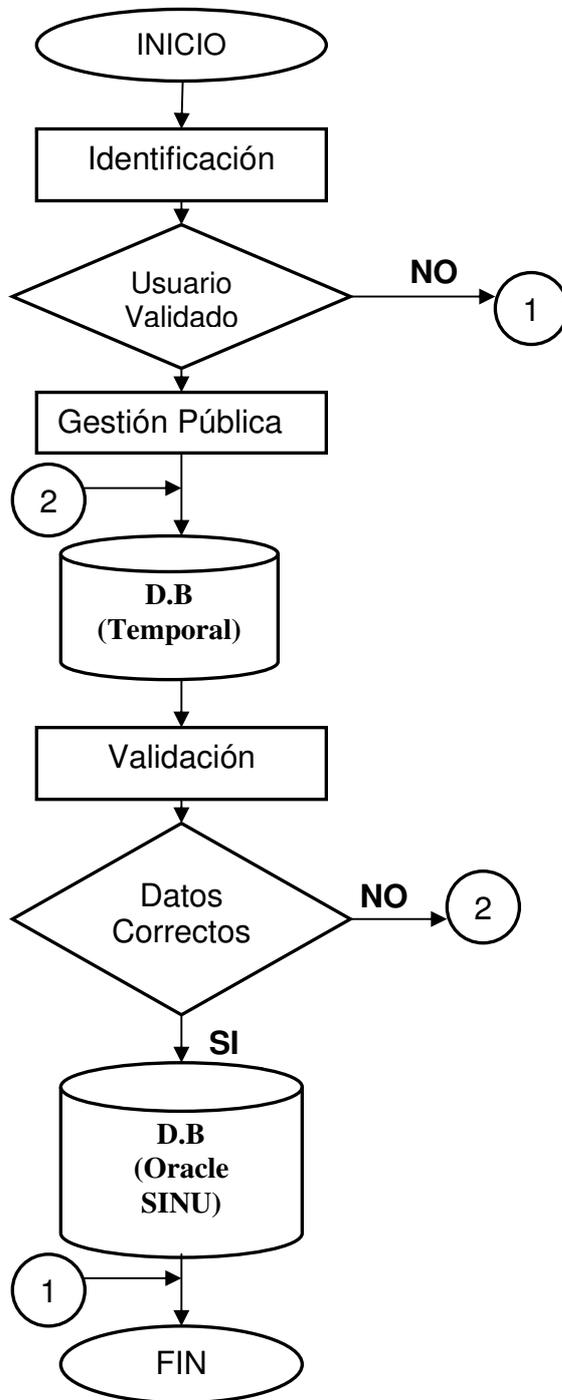


Gráfico No.1

DEFINICIÓN DEL GRAFICO

El sistema inicia con la **Identificación** de un usuario que desee ingresar, el sistema validara si el usuario esta identificado o no, en caso de no estar identificado el usuario no podrá ingresar y terminara su proceso en el sistema. Si el usuario es identificado podrá ingresar al sistema **Gestión Publica** y podrá modificar sus datos. Los datos serán guardados en una base de datos temporal. Estos datos que fueron almacenados en esta base de datos pasaran a un proceso de **validación**, con el fin de corroborar estos datos, si los datos son correctos serán almacenados en la base de datos principal (SINU).

OBJETIVO

Diseñar e implementar una interfaz de usuario que cumpla con las reglas de negocio implementadas y parametrizadas en SINU, con el fin de poblar la base de datos del mismo y obtener consultas para entregar información a los entes de control estatal y para uso interno de la Institución, en lo relacionado con egresados.

CONDICIONES BÁSICAS

Son los datos mínimos y básicos que todo egresado debe poseer, los cuales sirven para identificar y localizar al egresado. Cualquier actualización en los datos personales, será almacenada en una Base de Datos Temporal, para luego ser revisada por el validador, quien rectificará esta información y la almacenará en SINU, se llevará un histórico de los cambios y/o actualizaciones respectivas.

DATOS PERSONALES

Información personal del egresado. No podrán ser modificados por el mismo. En caso de existir algún dato erróneo, deberá dirigirse a Registro y Control donde modificarán este dato.

Número de Identificación: Este campo es de consulta, no se puede actualizar. En el caso que el documento de identificación se encuentre erróneo, el egresado deberá llevar su documento de identificación a Registro y Control donde modificarán este dato.

Si existe un cambio o actualización del Documento de Identidad, el cambio debe ser registrado y llevar su historial el cual será informado al SNIES.

Tipo de Identificación: Es un campo de consulta, no se puede actualizar.

Lugar de Expedición: Este campo es de consulta, y hace referencia al lugar de expedición del documento de identidad. No es modificable. En el caso que el Lugar de Expedición se encuentre erróneo, el egresado deberá llevar su documento de identificación a Registro y Control donde modificarán este dato.

Si el egresado encuentra su *Tipo de Documento* erróneo, deberá realizar el mismo procedimiento del cambio de Número de Identificación.

Primer Apellido: Es un campo de consulta, este campo es obligatorio para un egresado. Este dato es consultado de la base de datos, motivo por el cual no puede ser modificado. En el caso que el Primer Apellido se encuentre erróneo, el egresado deberá llevar su documento de identificación a Registro y Control donde modificarán este dato.

Segundo Apellido: Es un campo de consulta. Este dato es adquirido de la base de datos, motivo por el cual no puede ser modificado. En el caso que el Segundo Apellido se encuentre erróneo, el egresado deberá llevar su documento de identificación a Registro y Control donde modificarán este dato.

Nombres: Es un campo de consulta, este campo es obligatorio para un egresado. Este dato es adquirido de la base de datos, motivo por el cual no puede ser modificado. En el caso que los Nombres se encuentren erróneos, el egresado deberá llevar su documento de identificación a Registro y Control donde modificarán este dato.

Sexo: Es un campo que será consultado desde la base de datos, y no puede ser modificado. En el caso que el Sexo se encuentre erróneo, el egresado deberá llevar su documento de identificación a Registro y Control donde modificarán este dato.

Estado Civil: Es un campo tipo lista, el cual puede ser modificado por el egresado.

Natural: Es una consulta a la base de datos que muestra el tipo de persona que es (Natural o Jurídica) y no es modificable.

Fecha de Nacimiento: Este campo de consulta, en caso de estar erróneo el egresado deberá comunicarse con registro y control no puede ser modificado. En caso de estar erróneo el dato el egresado deberá acercarse a Registro y Control.

Grupo Sanguíneo: Este campo es de consulta, el cual me muestra los diferentes grupos sanguíneos. No puede ser modificado. En el caso que el Grupo Sanguíneo se encuentre erróneo, el egresado deberá llevar su documento de identificación a Registro y Control donde modificarán este dato.

Factor RH: Este campo invoca una ventana emergente la cual muestra el factor RH de una persona (Positivo o negativo). Este campo no puede ser modificado. En el caso que el Factor RH se encuentre erróneo, el egresado deberá llevar su documento de identificación a Registro y Control donde modificarán este dato.

Lugar de Nacimiento: Este campo es de consulta y no podrá ser modificado. En el caso que el Lugar de Nacimiento se encuentre erróneo, el egresado deberá llevar su documento de identificación a Registro y Control donde modificarán este dato.

Nacionalidad: Este campo es de consulta, puede ser actualizado; de ser así, este será almacenado en una tabla temporal para su validación. Una vez validado y autenticado será almacenado en la base de datos temporal.

Dependencia Econ: Este campo invoca una ventana emergente la cual muestra opciones de las posibles dependencias económicas de un egresado.

Num. Personas. Núcleo. Fam: Este campo es una caja de texto, solo debe permitir ingresar caracteres numéricos, no debe llevar espacios.

Personas a Cargo: Este campo es una caja de texto, solo debe permitir ingresar caracteres numéricos, no debe llevar espacios.

Lugar de Residencia: Este campo invoca una ventana emergente la cual muestra opciones de las diferentes ciudades de Colombia. Si la ciudad no existe, esta se podrá anexar una tabla temporal, después será validada y finalmente anexada a la base de datos.

Dirección Residencia: Este campo es de tipo texto y puede ser modificado por un egresado. Este campo es modificable, y se llevará un histórico de las actualizaciones.

Barrio: Campo de tipo lista, el cual contiene un listado de barrios, puede ser actualizado. Si el barrio no existe este podrá ser anexado a una tabla temporal mientras se valida el nuevo barrio, una vez hecho esto se podrá ingresar a la base de datos principal.

Teléfono Residencia: Campo de tipo texto, este campo es actualizable y solo debe permitir el ingreso de caracteres numéricos.

Celular: Este campo es una caja de tipo texto, actualizable y solo debe permitir el ingreso de caracteres numéricos.

E-mail: Este campo es de tipo texto y deberá ser validado.

E-mail secundario: Sería un e-mail distinto al de la Universidad.

Código Tercero: Código dado por la Universidad. No puede ser modificado. Este código no es de importancia para el egresado, por lo tanto éste no lo visualizará. Este código no se manejará.

Discapacidades: Es un botón el cual muestra las diferentes discapacidades físicas de una persona, mediante un menú de check boxes. Este campo puede ser modificado, en caso de ser modificado o no encontrar la incapacidad, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Para actualizar una discapacidad, el egresado deberá realizar el mismo proceso que se realizar cuando éste va a cambiar su número de Documento de Identificación.

Libreta militar numero: Este campo es una consulta a la base de datos y no debe permitir cambios. Este campo no es modificable. En el caso que el número de Libreta Militar se encuentre erróneo, el egresado deberá llevar su documento de identificación a Registro y Control donde modificarán este dato.

Distrito: Este campo es una consulta a la base de datos. Este campo no es modificable. En el caso que el Distrito se encuentre erróneo, el egresado deberá llevar su documento de identificación a Registro y Control donde modificarán este dato.

Nombre EPS: Este campo es una consulta y muestra las diferentes entidades de salud a las que puede estar afiliado un egresado. En caso de que el egresado se encuentre vinculado laboralmente con la Fundación Universitaria del Área Andina, no puede ser modificado; en caso de que exista información errónea o no encontrar el Nombre de EPS, deberá dirigirse a la División de Asuntos Laborales. Si el egresado no se encuentra vinculado laboralmente con la Fundación Universitaria del Área Andina y existe algún error en el nombre de la EPS, éste podrá guardar y/o modificar esta información, la cual será guardada en una tabla temporal para la validación y autenticación, para luego ser guardada en la base de datos principal.

Estrato Socio Económico: Este campo es de tipo texto, solo debe admitir caracteres numéricos.

Salarios Mínimos: Este campo invoca una ventana emergente la cual muestra opciones de cuantos salarios mínimos gana una persona y devuelve un código de 1 al 4. Los parámetros utilizados por SINU son los siguientes:

- | | |
|---|------------|
| 1 | Menos de 3 |
| 2 | De 4 y 6 |
| 3 | De 7 y 10 |
| 4 | Más de 10 |

Estudios en la Institución

Información sobre los estudios realizados por un egresado en la Fundación Universitaria del Área Andina. Si se presenta Información errónea en alguno de los siguientes campos, se debe sustentar el cambio en Registro y Control.

Programa: Este campo no puede ser modificado, porque es una consulta a la base de datos.

Pensum: Este campo es de tipo texto y no podrá ser modificado, porque es una consulta a la base de datos.

Per. Ingreso: Este campo es de tipo texto y no podrá ser modificado, porque es una consulta a la base de datos.

Código: Este campo es de tipo texto y no podrá ser modificado, porque es una consulta a la base de datos.

Fecha Ingreso: Este campo es de tipo texto y no podrá ser modificado, porque es una consulta a la base de datos.

Fecha Retiro: Este campo es de tipo texto y no podrá ser modificado, porque es una consulta a la base de datos.

Grupo: Este campo es de tipo texto y no podrá ser modificado, porque es una consulta a la base de datos.

Estado: Este campo es de tipo texto y no podrá ser modificado, porque es una consulta a la base de datos.

Otros Datos: Este opción es un texto de area en la cual se digitan caracteres de a – z, cuyo fin es el de complementar algún dato que el egresado crea importante, este campo tendrá un limite de 640 caracteres.

ESTUDIOS

Secundaria

En el caso en que el registro del colegio vacío, éste se podrá ingresar; en caso de que exista algún dato erróneo el egresado deberá acercarse a Registro y Control.

Institución: Este campo invoca una ventana emergente la cual muestra las diferentes instituciones educativas del país. Si la Institución no aparece en dicho listado, esta nueva se podrá ingresar al listado, mediante una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Código: Es una consulta a la base de datos, solo se podrá visualizar por parte del egresado, esta consulta hace referencia al código de la institución.

SNP: Este campo es una consulta. Este campo no puede ser modificado. En el caso que el SNP se encuentre erróneo, el egresado deberá llevar su documento a Registro y Control donde modificarán este dato.

Jornada: Este campo es de tipo lista y muestra las diferentes jornadas de estudio.

Calendario: Este campo es de tipo lista y muestra el tipo de calendario de un estudiante (Calendario A, Calendario B, Calendario C).

Población: Este campo es de tipo lista y muestra el tipo de población. Si la Población no aparece en dicho listado, esta nueva se podrá ingresar al listado, mediante una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Título Obtenido: Este campo es de tipo texto, solo debe permitir ingreso caracteres de a – z.

Año Grado: Este campo es una caja de texto de tipo String y solo debe aceptar caracteres numéricos.

Fecha Terminó: Este campo es una caja de texto de tipo String, no puede ser modificado.

Es Validación: Este campo es un radio botón (si/no).

Grado de Honor: Este campo es un radio botón (si/no).

Observaciones: Este campo es una caja de texto de tipo String.

Universitarios

Estos estudios Universitarios son realizados en otras Instituciones diferentes a la Fundación Universitaria del Área Andina y puede ser actualizado por el egresado.

Institución: Este campo invoca una ventana emergente la cual muestra las diferentes instituciones de educación superior del país y no es modificable porque será un dato que es llamado desde la base de datos. En caso de ser modificado o no encontrar la Institución, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Título Obtenido: Este campo es una consulta a la base de datos, solo debe permitir ingreso caracteres de a – z. En caso de ser modificado o no encontrar el Título Obtenido, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Fecha Terminación: Este campo es una consulta a la base de datos. En caso de ser modificado o no encontrar la Fecha de Terminación, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Año de terminación: Este campo es una consulta a la base de datos. En caso de ser modificado o no encontrar el Año de Terminación, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Modalidad: Es un campo tipo lista, es una consulta a la base de datos. En caso de ser modificado o no encontrar la Modalidad, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Profesión: Este campo invoca una ventana emergente la cual muestra las diferentes profesiones que puede tener un egresado. En caso de ser modificado o no encontrar la Profesión, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Nivel de formación: Este campo invoca una ventana emergente la cual muestra el nivel de información que tiene el egresado. En caso de ser modificado o no encontrar la incapacidad, se guardarán los cambios realizados en una tabla

temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Registro de título: Es una consulta a la base de datos. En caso de ser modificado o no encontrar el Registro de Título, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Numero Tarjeta Profesional: Es una consulta a la base de datos. En caso de ser modificado o no encontrar el Número de Tarjeta Profesional, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Duración (semestre): Este campo es una consulta a la base de datos y solo debe cargar caracteres numéricos. En caso de ser modificado o no encontrar la Duración, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Estudios Complementarios:

Estudios realizados por el egresado de educación continuada o actualización, y podrán ser modificados por el egresado.

Nombre del Curso: Este es un area de texto de tipo string.

Número de Horas: Este campo es una caja de texto de tipo String y solo acepta caracteres de tipos numéricos.

Institución: Este campo invoca una ventana emergente la cual muestra las diferentes instituciones de educación superior del país.

Fecha de Terminación: Este campo es una caja de texto de tipo String, no puede ser modificado.

Esta Relacionado con la Profesión: Este campo es un radio botón (si/no).

LABORAL

Experiencia Laboral del egresado. Podrá ser modificado por el mismo.

Entidad: Este campo invoca una ventana emergente la cual muestra las diferentes entidades donde puede estar vinculado un egresado laboralmente. Este campo puede ser modificado, en caso de ser modificado o no encontrar la Entidad, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Nivel del cargo: Es una caja de texto de tipo string, puede ser modificado. Este campo puede ser modificado, en caso de ser modificado o no encontrar el Nivel del Cargo, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Dependencia: Es una caja de texto de tipo string, puede ser modificado.

Sede (Sucursal): Es una caja de texto de tipo string, puede ser modificado.

Cargo (Nivel): Este campo invoca una ventana emergente la cual muestra los diferentes niveles del cargo y el respectivo nombre. Estos son los niveles de cargo que puede tener una persona según SINU:

- 1 Directivo
- 2 Profesional
- 3 Auxiliar
- 4 Servicios
- 5 Empleado
- 6 Propietario

Ingreso: Este campo es una caja de texto de tipo string, solo debe recibir caracteres de tipo numérico y no debe llevar espacios.

Retiro: Este campo es una caja de texto de tipo String, no puede ser modificado.

Tiempo: Este campo es una caja de texto de tipo String, no puede ser modificado.

Vinculación: Este campo invoca una ventana emergente la cual muestra los diferentes tipos de vinculación. SINU ofrece las siguientes opciones:

1. Tiempo Completo.
2. Tiempo Parcial.
3. Hora Cátedra.
4. A. D. Honores.

En caso de ser modificado o no encontrar la Vinculación, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Estado de Vinculación: Este campo es un radio botón (Inactivo/Activo)

Clase de Vinculación: Este campo es un radio botón (Empleado/Propietario).

Remuneración: Este campo es una caja de texto de tipo string, solo debe recibir caracteres de tipo numérico y no debe llevar espacios.

Tipo de Empresa: Este campo es un radio botón (Privada/Oficial).

Actividades realizadas: Este campo es una area de texto de tipo String.

VINCULO CON LA INSTITUCIÓN

Dependencia: Es una consulta a la base de datos. En caso de ser modificado o no encontrar el Número de Tarjeta Profesional, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Programa Académico: Es una consulta a la base de datos. En caso de ser modificado o no encontrar el Número de Tarjeta Profesional, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Cargo: Es una consulta a la base de datos. En caso de ser modificado o no encontrar el Número de Tarjeta Profesional, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Fecha Ingreso: Este campo es una consulta a la base de datos y no podrá ser modificado.

Fecha Retiro: Este campo es una consulta a la base de datos y no podrá ser modificado.

Remuneración: Este campo es una consulta a la base de datos y no podrá ser modificado.

Atributos Laboral: Este campo es una consulta a la base de datos y no podrá ser modificado.

Estado Actual: Este campo es una consulta a la base de datos y no podrá ser modificado.

EXPERIENCIA DOCENTE

Señala las actividades del docente en otras instituciones.

Institución: Este campo es una consulta a la base de datos y no podrá ser modificado.

Asignaturas: Este campo es una consulta a la base de datos y no podrá ser modificado.

Niveles de Cargo: Este campo invoca una ventana emergente la cual muestra los diferentes niveles que pudo tener un docente en otras instituciones. Los niveles según SINU son:

- 1 Directivo
- 2 Profesional
- 3 Auxiliar
- 4 Servicios
- 5 Empleado
- 6 Propietario

Este campo puede ser modificado, en caso de ser modificado o no encontrar la Entidad, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Facultad – Programa – Dependencia: Este campo es una consulta a la base de datos y no podrá ser modificado.

Actividades Realizadas: Este es un campo de texto, en el cual se almacenara las actividades realizadas de un docente. Este campo puede ser modificado, en caso de ser modificado, se guardarán los cambios realizados en una tabla temporal para la validación y autenticación, para luego ser guardado en la base de datos principal.

Fecha Ingreso: Este campo es una consulta a la base de datos y no podrá ser modificado.

Fecha Retiro: Este campo es una consulta a la base de datos y no podrá ser modificado.

Tipo de Vinculación: Este campo invoca una ventana emergente la cual muestra los diferentes tipos de vinculación. SINU ofrece las siguientes opciones:

5. Tiempo Completo.
6. Tiempo Parcial.
7. Hora Cátedra.
8. A. D. Honores.

IDIOMAS

Dominios de Otros Idiomas

Al modificar alguno de estos cambios, se llevará el control histórico, de cuantas veces y que campos ha modificado.

Idioma: Es una caja de texto de tipo String la cual solo debe aceptar caracteres de la a – z.

Lee: Este campo es un radio botón (si/no)

Escribe: Este campo es un radio botón (si/no)

Habla: Este campo es un radio botón (si/no)

ANÁLISIS DE PROCEDIMIENTOS

1. Información no encontrada en las listas de valores

El egresado podrá agregar o adicionar la información deseada en los siguientes campos:

INFORMACIÓN PERSONAL

1. Lugar de Expedición (Cédula)
2. Lugar de Nacimiento
3. Nacionalidad
4. Dependencia Económica
5. Lugar Residencia
6. Barrio
7. Distrito
8. Nombre EPS
9. Salarios Mínimos
10. Discapacidades

ESTUDIOS

11. Institución

UNIVERSITARIOS

12. Institución
13. Profesión
14. Nivel de Formación

ESTUDIOS COMPLEMENTARIOS

15. Institución

LABORAL

16. Entidad
17. Niveles de Cargo
18. Tipos de Vinculación

EXPERIENCIA DOCENTE

19. Niveles de Cargo
20. Institución
21. Tipo de Vinculación

Esta información será almacenada en una tabla temporal, para luego ser validada por un tercero vinculado con la Fundación Universitaria del Área

Andina, que deberá verificar los datos, una vez validados y aceptados los datos que ingresó el egresado serán trasladados a la base de datos SINU.

2. Ingreso a Gestión Pública Hoja de Vida de Egresados

El egresado que desee ingresar deberá ser validado por el sistema SINU, mediante su usuario y contraseña del correo institucional, los cuales le fueron entregados al inicio del programa académico, en el departamento o programa respectivo, en un sobre, de acuerdo a la normatividad vigente; para luego ser validado contra LDAP. Si el sistema no reconoce al egresado no podrá acceder al sistema.

3. Actualización de los Datos de la Gestión Pública Hoja de Vida de Egresados

Una vez validado el egresado y si se permitió su ingreso, éste podrá ver sus datos y modificarlos, mediante cuatro formularios:

1. Datos Personales
2. Estudios
3. Información Laboral
4. Idiomas

Estos datos quedan almacenados en una base de datos temporal. Gestión Pública Hoja de Vida de Egresados, le mostrará un mensaje al egresado, en el cual se avisa que los datos actualizados por este, no se actualizarán en línea, sino que transcurrirá un tiempo mientras estos son validados.

4. Validación

El proceso de validación consta de la confirmación y actualización de los datos del egresado. Se tienen dos roles en cuanto a la seguridad de Gestión Pública Hoja de Vida de Egresados, el primero, el cual modificará los datos de Información Laboral, si el egresado se encuentra actualmente vinculado a la universidad; el segundo, actualizará los demás egresados. Para la validación, la persona encargada, ingresará con su usuario y contraseña, y observará un listado de las últimas actualizaciones de datos realizadas por el egresado en Gestión Pública Hoja de Vida del Egresado, para su respectiva rectificación; el validador podrá ver los datos completos del egresado mediante un formulario, el cual señalará los últimos campos modificados por el egresado, para facilitar la función del validador, quien aprobará o no esta información. Después de ser

revisada la información y modificada correctamente, según el estándar definido por la universidad, ésta será almacenada en SINU, en caso de ser correcta; en caso contrario, se le notificará al Egresado sobre la información inconsistente.

5. Visualizar la Información del Egresado

Después de que el validador confirme los datos actualizados en la Base de Datos Temporal, y migre estos datos a SINU, el egresado podrá visualizar sus datos actualizados.

MANUAL DEL USUARIO

OBJETO

Diseñar e implementar una interfaz de usuario que cumpla con las reglas de negocio implementadas y parametrizadas en Sinu, con el fin de poblar la base de datos del mismo y obtener consultas para entregar información a los entes de control estatal y para uso interno de la Institución, en lo relacionado con egresados.

ALCANCE

De acuerdo con las políticas, normatividad y procesos vigentes, la actividad de ingreso y actualización de la información de la Hoja de Vida de los Egresados en el Sistema de Información Sinu, consiste en el registro de la información personal, laboral y académica por el egresado vía web.

PROCESO

1. Ingreso

El ingreso de la información corresponde únicamente al Egresado, quien ha cursado y aprobado satisfactoriamente todas las asignaturas o créditos que componen el plan de estudios reglamentado para un programa académico de la Fundación Universitaria del Área Andina.

Para realizar el ingreso, el Egresado debe tener su usuario y clave, los cuales le fueron entregados al inicio del programa académico, en el departamento o programa respectivo, en un sobre, de acuerdo a la normatividad vigente.

Para acceder al sistema de Información y realizar el registro de la información, el Egresado debe ingresar al portal institucional (www.areandina.edu.co), identificarse ante el sistema con el usuario y clave de acceso asignado y entregado en el sobre y seguir las instrucciones correspondientes.

Cuando se abre la página de Ingreso de la Información, el Egresado procederá a registrar su Información Personal.

Nota 1: Se recomienda que una vez digitada se verifique antes de guardar, para revisar y corregir errores de transcripción o digitación.

Nota 2: Por seguridad, al terminar el proceso de registro y guardado de la información, el Egresado procederá a cerrar su sesión, con el fin de evitar que un tercero manipule su información.

2. Actualización

El Egresado podrá ver y actualizar sus datos, realizando el mismo proceso para ingresar su información.

Nota 3: Gestión Pública Hoja de Vida de Egresados, le mostrará un mensaje al egresado, en el cual se avisa que los datos actualizados por este, no se actualizarán instantáneamente, sino que transcurrirá un tiempo mientras estos son validados.

3. Validación

El proceso de validación consta de la confirmación y actualización de los datos del egresado. Se tienen dos roles en cuanto a la seguridad de Gestión Pública Hoja de Vida de Egresados, el primero, el cual modificará los datos de Información Laboral, si el egresado se encuentra actualmente vinculado a la universidad; el segundo, actualizará los demás egresados. Para la validación, la persona encargada, ingresará con su usuario y contraseña, y observará un listado de las últimas actualizaciones de datos realizadas por el egresado en Gestión Pública Hoja de Vida del Egresado, para su respectiva rectificación; el validador podrá observar los datos completos del egresado, quien aprobará o no esta información. Después de ser revisada la información y modificada correctamente, según el estándar definido por la universidad, ésta será almacenada, en caso de ser correcta; en caso contrario, se le notificará al Egresado sobre la información inconsistente.

4. Seguridad

Por la naturaleza, importancia y consecuencias legales, es necesario que los involucrados en el proceso de ingreso y validación de la información observen y cumplan los siguientes lineamientos.

El usuario y clave de acceso es personal e intransferible, es decir, solo debe conocerlo el Egresado. Por lo anterior, al recibir el sobreflex que contiene el usuario y clave de acceso, debe verificarse y asegurarse que no ha sido abierto, en caso contrario debe colocar en conocimiento de la situación al director de la División de Sistemas e Informática para tomar las acciones correctivas necesarias. Así mismo, tampoco debe entregarse a ningún funcionario de la Institución o persona alguna.

Cambiar periódicamente la clave de acceso. Para realizarlo, desde la página de la Institución en la pantalla de Mi Correo, entrar y dar click en Preferencias del Menú General y luego en “Cambiar su Contraseña” y seguir las instrucciones que allí aparecen.

El usuario y clave de acceso es único para cada Egresado independientemente del departamento o programa académico donde sea egresado.

5. Visualizar la Información del Egresado

Después de que el validador confirme los datos actualizados, el egresado podrá visualizar sus datos actualizados.

MANUAL DE CONFIGURACIÓN DE LAS HERRAMIENTAS DE DESARROLLO WEB

FUAA
FUNDACIÓN UNIVERSITARIA DEL ÁREA ANDINA

Autores:	Jorge Andrés Cañas Vargas Walter Julián Bernal Suárez
Fecha Creación:	04/05/2007
Ultima Actualización:	04/05/2007
Código:	MAN_CONFIG_WEB_V1.0
Versión:	1.0

Control de Proyecto

Manual:	Configuración de las herramientas de desarrollo Web
Versión:	1.0
Proyecto:	GESTIÓN PÚBLICA HOJA DE VIDA DE EGRESADOS
Realizado Por:	Jorge Andrés Cañas Vargas. Walter Julián Bernal Suárez. Mail : jocanas@areandina.edu.co wabernal3@areandina.edu.co

Registro de Cambios

Fecha	Autor	Versión	Descripción de los Cambios
04/05/07	Jorge Andrés Cañas V. Walter Julián Bernal S.	1.0	Creación

INTRODUCCIÓN

El siguiente manual contiene una serie de pasos para configurar adecuadamente el entorno de desarrollo.

Sistemas Operativos:

- Windows 2000 / XP.

CONFIGURACIÓN DEL ENTORNO DE DESARROLLO

El siguiente manual contiene una serie de pasos para configurar adecuadamente el entorno de desarrollo.

Sistemas Operativos:

- Windows 2000 / XP.

Requerimientos Mínimos:

- Instalación del TOMCAT



jakarta-tomcat-5.0.28
Apache Tomcat Installer
Apache Software Foundation

Nota: El Tomcat fue entregado por parte de ACIES Ltda., este se copia y pega dentro del directorio *Software*; previamente se configura el puerto.

Debe realizar los siguientes pasos:

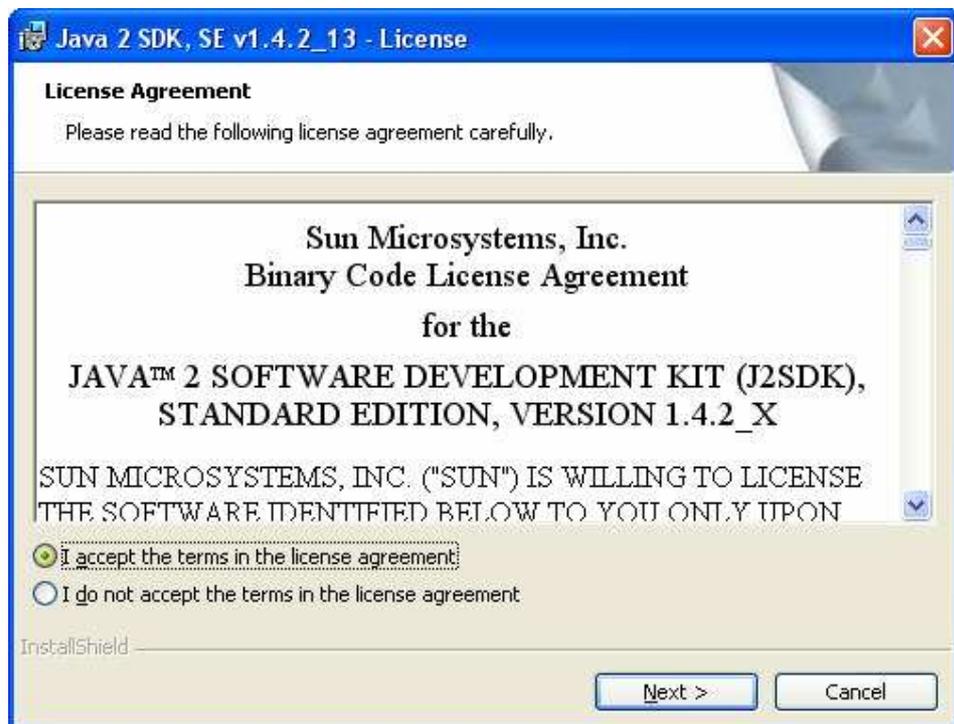
Defina el directorio en el cual va a instalar los instaladores

Se debe crear un directorio llamado **SOFTWARE** en nuestro root.

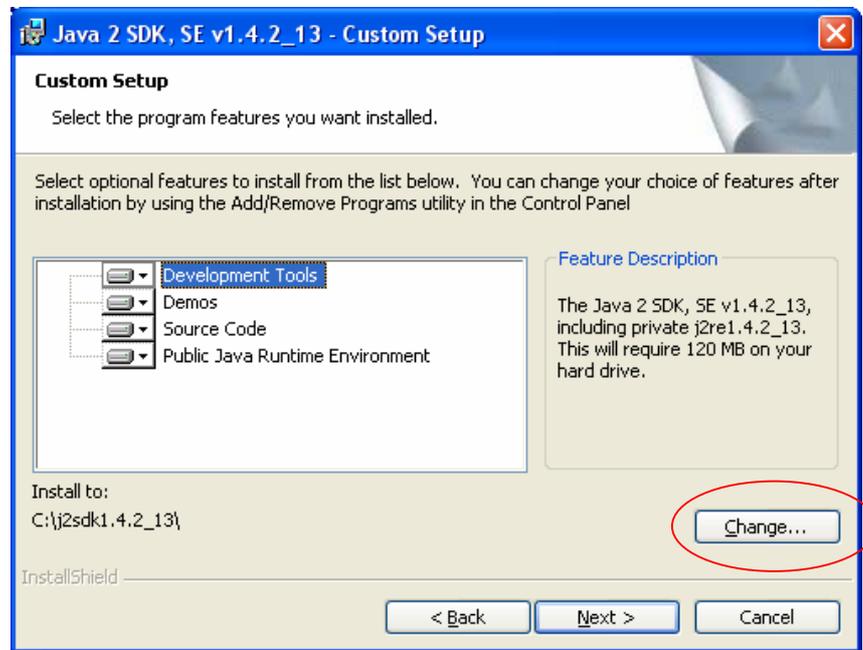


instalar la JDK.

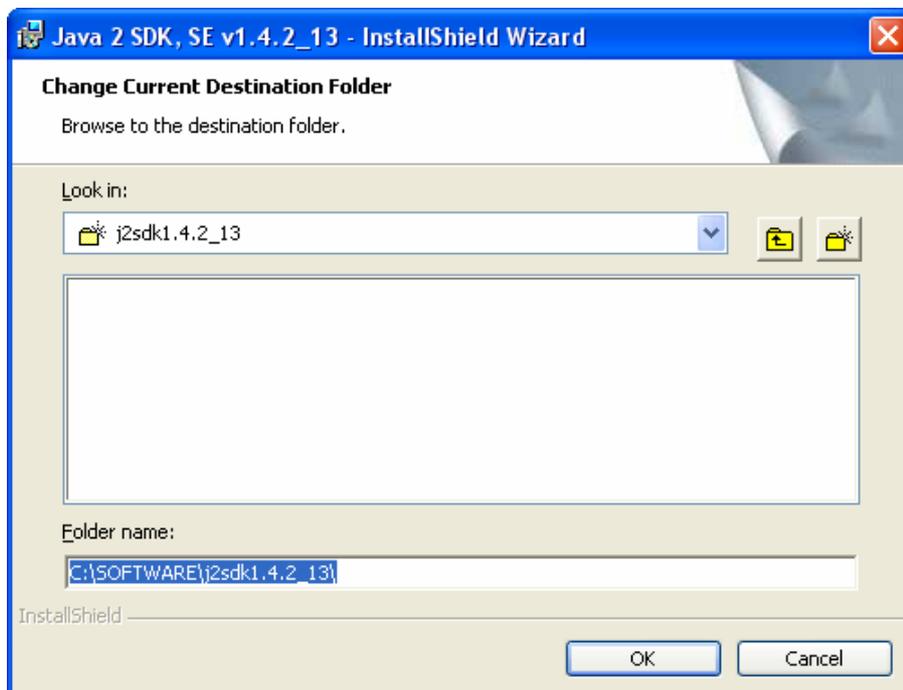
Descargar (<http://java.sun.com/javase/downloads/index.jsp>) y ejecutar el instalador.



Ubicación del directorio de instalación de la JDK:



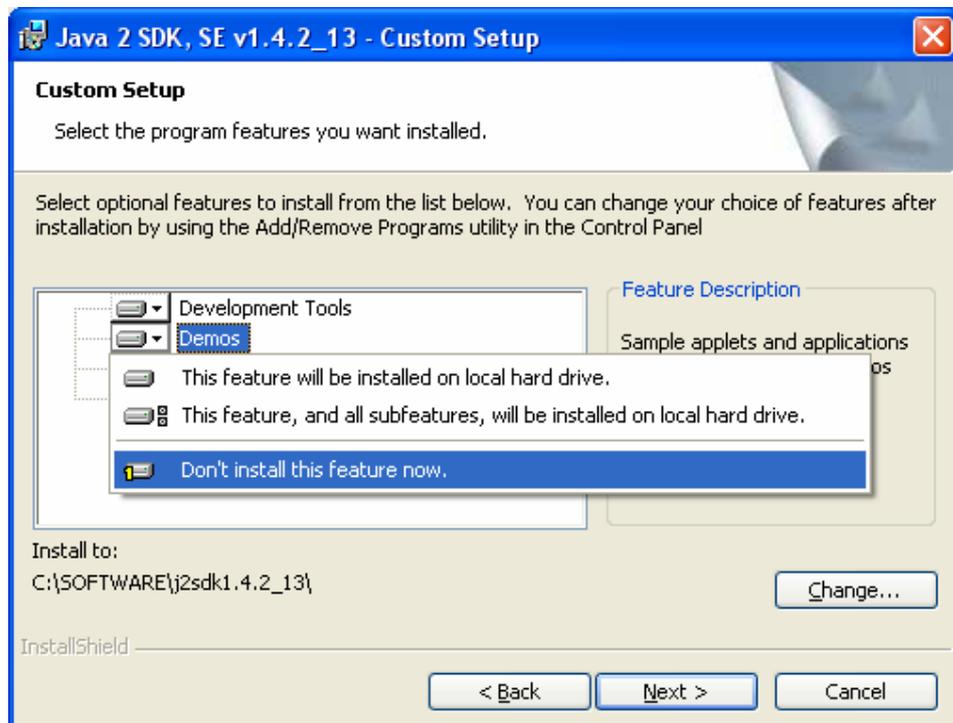
Entramos en Change...



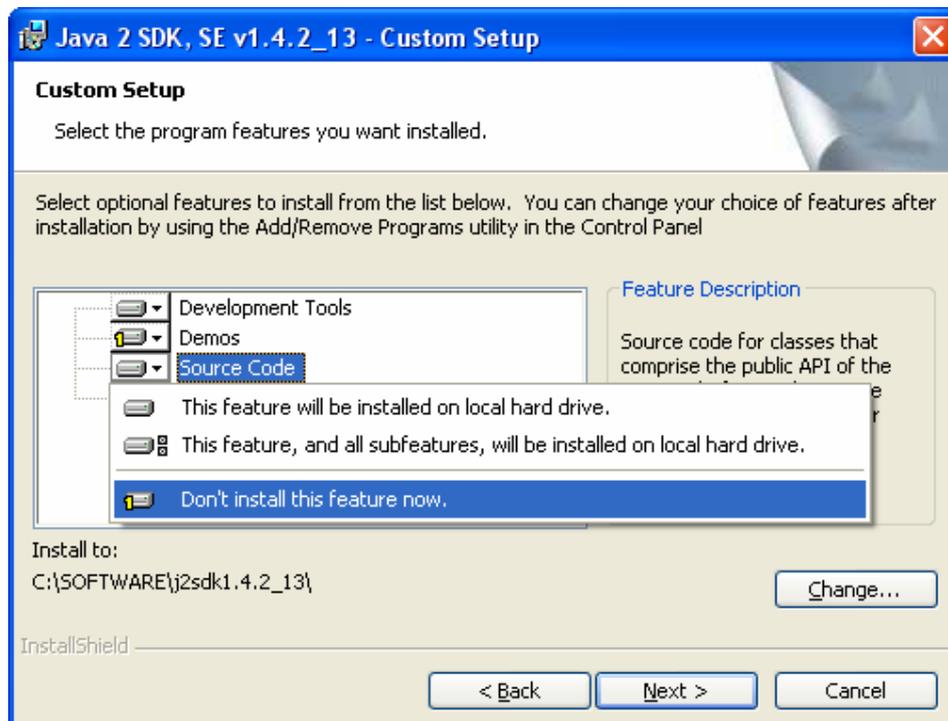
Presionamos OK.

Cancelar la instalación de los módulos que no se usarán.

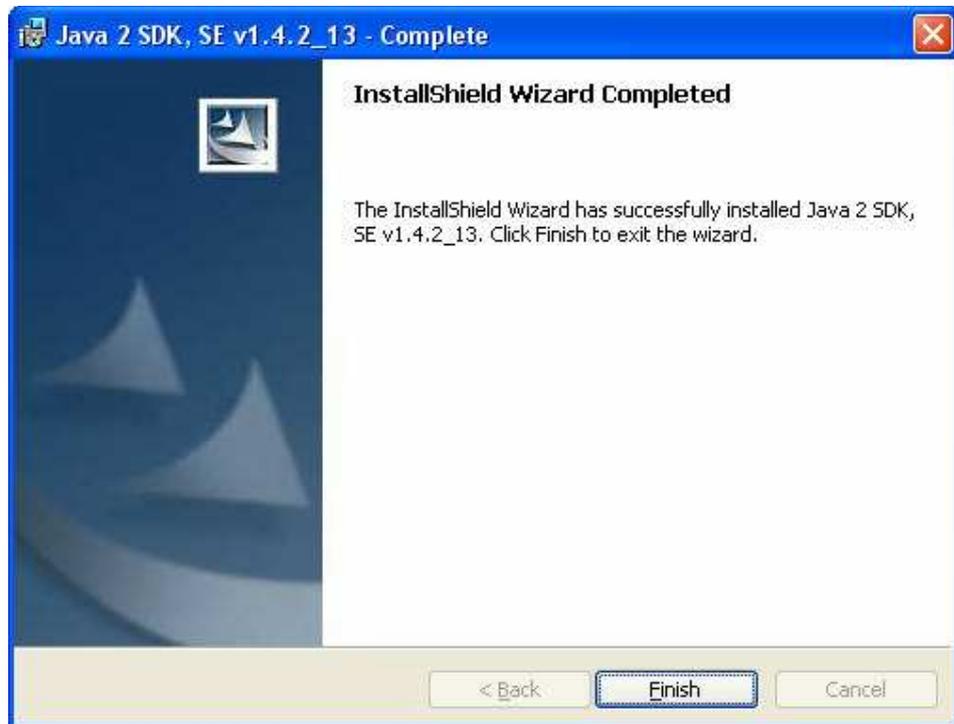
- Demos



- Source Code



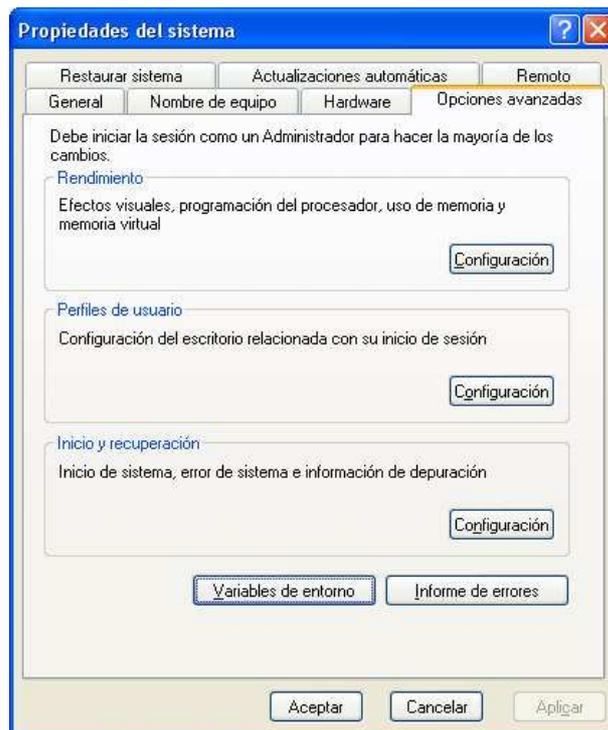
Finalizar



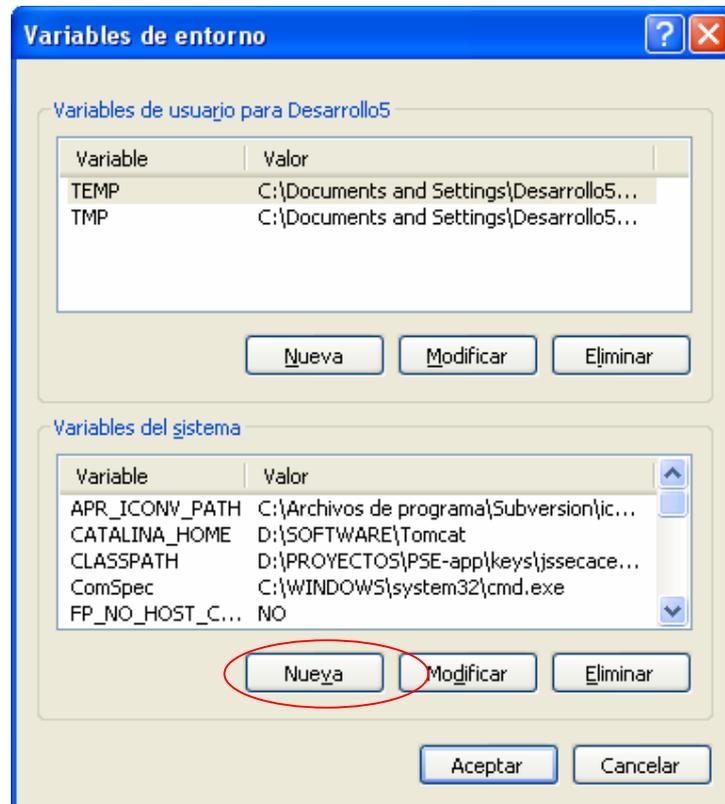
Definir la variable de entorno **JAVA_HOME**



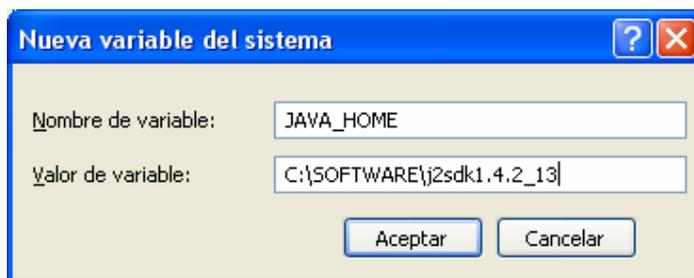
En Propiedades del Sistema



En Variables de Entorno



Entramos en Nueva, y definimos la variable de entorno



Configuración Tomcat

En C:\SOFTWARE\Tomcat\conf\Server.xml

Se configura el Server.xml del Tomcat de la siguiente forma, (sobrescribimos la siguiente información):

```
<!-- Example Server Configuration File -->
<!-- Note that component elements are nested corresponding to their
parent-child relationships with each other -->

<!-- A "Server" is a singleton element that represents the entire JVM,
which may contain one or more "Service" instances. The Server
listens for a shutdown command on the indicated port.

Note: A "Server" is not itself a "Container", so you may not
define subcomponents such as "Valves" or "Loggers" at this level.
-->

<Server port="8005" shutdown="SHUTDOWN">

  <!-- Comment these entries out to disable JMX MBeans support used for the
administration web application -->
  <Listener className="org.apache.catalina.mbeans.ServerLifecycleListener"
/>
  <Listener
className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener" />
  <Listener
className="org.apache.catalina.storeconfig.StoreConfigLifecycleListener"/>

  <!-- Global JNDI resources -->
  <GlobalNamingResources>

    <!-- Test entry for demonstration purposes -->
    <Environment name="simpleValue" type="java.lang.Integer" value="30"/>

    <!-- Editable user database that can also be used by
    UserDatabaseRealm to authenticate users -->
    <Resource name="UserDatabase" auth="Container"
type="org.apache.catalina.UserDatabase"
description="User database that can be updated and saved"
factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
pathname="conf/tomcat-users.xml" />
```

```
</GlobalNamingResources>
```

```
<!-- A "Service" is a collection of one or more "Connectors" that share
a single "Container" (and therefore the web applications visible
within that Container). Normally, that Container is an "Engine",
but this is not required.
```

```
    Note: A "Service" is not itself a "Container", so you may not
    define subcomponents such as "Valves" or "Loggers" at this level.
```

```
-->
```

```
<!-- Define the Tomcat Stand-Alone Service -->
<Service name="Catalina">
```

```
    <!-- A "Connector" represents an endpoint by which requests are received
and responses are returned. Each Connector passes requests on to the
associated "Container" (normally an Engine) for processing.
```

```
    By default, a non-SSL HTTP/1.1 Connector is established on port 8080.
    You can also enable an SSL HTTP/1.1 Connector on port 8443 by
    following the instructions below and uncommenting the second Connector
    entry. SSL support requires the following steps (see the SSL Config
    HOWTO in the Tomcat 5 documentation bundle for more detailed
    instructions):
```

```
    * If your JDK version 1.3 or prior, download and install JSSE 1.0.2 or
    later, and put the JAR files into "$JAVA_HOME/jre/lib/ext".
```

```
    * Execute:
```

```
        %JAVA_HOME%\bin\keytool -genkey -alias tomcat -keyalg RSA
```

```
(Windows)
```

```
        $JAVA_HOME/bin/keytool -genkey -alias tomcat -keyalg RSA (Unix)
    with a password value of "changeit" for both the certificate and
    the keystore itself.
```

```
    By default, DNS lookups are enabled when a web application calls
    request.getRemoteHost(). This can have an adverse impact on
    performance, so you can disable it by setting the
    "enableLookups" attribute to "false". When DNS lookups are disabled,
    request.getRemoteHost() will return the String version of the
    IP address of the remote client.
```

```
-->
```

```
<!-- Define a non-SSL HTTP/1.1 Connector on port 8080 -->
```

```
<Connector port="8087" maxHttpHeaderSize="8192"
    maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
    enableLookups="false" redirectPort="8443" acceptCount="100"
```

```
        connectionTimeout="20000" disableUploadTimeout="true" />
<!-- Note : To disable connection timeouts, set connectionTimeout value
to 0 -->
```

<!-- Note : To use gzip compression you could set the following properties :

```
        compression="on"
        compressionMinSize="2048"
        noCompressionUserAgents="gozilla, traviata"
        compressableMimeType="text/html,text/xml"
```

-->

```
<!-- Define a SSL HTTP/1.1 Connector on port 8443 -->
```

```
<!--
```

```
<Connector port="8443" maxHttpHeaderSize="8192"
        maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
        enableLookups="false" disableUploadTimeout="true"
        acceptCount="100" scheme="https" secure="true"
        clientAuth="false" sslProtocol="TLS" />
```

-->

```
<!-- Define an AJP 1.3 Connector on port 8009 -->
```

```
<Connector port="8009"
        enableLookups="false" redirectPort="8443" protocol="AJP/1.3" />
```

```
<!-- Define a Proxied HTTP/1.1 Connector on port 8082 -->
```

```
<!-- See proxy documentation for more information about using this. -->
```

```
<!--
```

```
<Connector port="8082"
        maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
        enableLookups="false" acceptCount="100"
connectionTimeout="20000"
        proxyPort="80" disableUploadTimeout="true" />
```

-->

```
<!-- An Engine represents the entry point (within Catalina) that processes
every request. The Engine implementation for Tomcat stand alone
analyzes the HTTP headers included with the request, and passes them
on to the appropriate Host (virtual host). -->
```

```
<!-- You should set jvmRoute to support load-balancing via AJP ie :
```

```
<Engine name="Standalone" defaultHost="localhost" jvmRoute="jvm1">
```

-->

```
<!-- Define the top level container in our container hierarchy -->
```

```

<Engine name="Catalina" defaultHost="localhost">

  <!-- The request dumper valve dumps useful debugging information about
  the request headers and cookies that were received, and the response
  headers and cookies that were sent, for all requests received by
  this instance of Tomcat. If you care only about requests to a
  particular virtual host, or a particular application, nest this
  element inside the corresponding <Host> or <Context> entry instead.

  For a similar mechanism that is portable to all Servlet 2.4
  containers, check out the "RequestDumperFilter" Filter in the
  example application (the source for this filter may be found in
  "$CATALINA_HOME/webapps/examples/WEB-INF/classes/filters").

  Request dumping is disabled by default. Uncomment the following
  element to enable it. -->
  <!--
  <Valve className="org.apache.catalina.valves.RequestDumperValve"/>
  -->

  <!-- Because this Realm is here, an instance will be shared globally -->

  <!-- This Realm uses the UserDatabase configured in the global JNDI
  resources under the key "UserDatabase". Any edits
  that are performed against this UserDatabase are immediately
  available for use by the Realm.
  <Realm className="org.apache.catalina.realm.UserDatabaseRealm"
  resourceName="UserDatabase"/>
  -->
  <Realm className="org.apache.catalina.realm.JAASRealm"
  appName="josso"
  userClassNames="org.josso.gateway.identity.service.BaseUserImpl"
  roleClassNames="org.josso.gateway.identity.service.BaseRoleImpl"
  debug="1" />
  <!-- Comment out the old realm but leave here for now in case we
  need to go back quickly -->
  <!--
  <Realm className="org.apache.catalina.realm.MemoryRealm" />
  -->

  <!-- Replace the above Realm with one of the following to get a Realm
  stored in a database and accessed via JDBC -->

  <!--
  <Realm className="org.apache.catalina.realm.JDBCRealm"
  driverName="org.gjt.mm.mysql.Driver"

```

```

        connectionURL="jdbc:mysql://localhost/authority"
        connectionName="test" connectionPassword="test"
        userTable="users" userNameCol="user_name"
userCredCol="user_pass"
        userRoleTable="user_roles" roleNameCol="role_name" />
-->

<!--
<Realm className="org.apache.catalina.realm.JDBCRealm"
        driverName="oracle.jdbc.driver.OracleDriver"
        connectionURL="jdbc:oracle:thin:@ntserver:1521:ORCL"
        connectionName="scott" connectionPassword="tiger"
        userTable="users" userNameCol="user_name"
userCredCol="user_pass"
        userRoleTable="user_roles" roleNameCol="role_name" />
-->

<!--
<Realm className="org.apache.catalina.realm.JDBCRealm"
        driverName="sun.jdbc.odbc.JdbcOdbcDriver"
        connectionURL="jdbc:odbc:CATALINA"
        userTable="users" userNameCol="user_name"
userCredCol="user_pass"
        userRoleTable="user_roles" roleNameCol="role_name" />
-->

<!-- Define the default virtual host
    Note: XML Schema validation will not work with Xerces 2.2.
-->
<Host name="localhost" appBase="webapps"
    unpackWARs="true" autoDeploy="true"
    xmlValidation="false" xmlNamespaceAware="false">

    <!-- Defines a cluster for this node,
        By defining this element, means that every manager will be changed.
        So when running a cluster, only make sure that you have webapps in
there
        that need to be clustered and remove the other ones.
        A cluster has the following parameters:

        className = the fully qualified name of the cluster class

        name = a descriptive name for your cluster, can be anything

        mcastAddr = the multicast address, has to be the same for all the
nodes

```

mcastPort = the multicast port, has to be the same for all the nodes

mcastBindAddr = bind the multicast socket to a specific address

mcastTTL = the multicast TTL if you want to limit your broadcast

mcastSoTimeout = the multicast readtimeout

mcastFrequency = the number of milliseconds in between sending a "I'm alive" heartbeat

mcastDropTime = the number a milliseconds before a node is considered "dead" if no heartbeat is received

tcpThreadCount = the number of threads to handle incoming replication requests, optimal would be the same amount of threads as nodes

tcpListenAddress = the listen address (bind address) for TCP cluster request on this host,
in case of multiple ethernet cards.
auto means that address becomes
InetAddress.getLocalHost().getHostAddress()

tcpListenPort = the tcp listen port

tcpSelectorTimeout = the timeout (ms) for the Selector.select() method in case the OS
has a wakeup bug in java.nio. Set to 0 for no timeout

printToScreen = true means that managers will also print to std.out

expireSessionsOnShutdown = true means that

useDirtyFlag = true means that we only replicate a session after setAttribute,removeAttribute has been called.
false means to replicate the session after each request.
false means that replication would work for the following piece of code: (only for SimpleTcpReplicationManager)

```
<%
HashMap map = (HashMap)session.getAttribute("map");
map.put("key","value");
%>
```

replicationMode = can be either 'pooled', 'synchronous' or 'asynchronous'.

* Pooled means that the replication happens using several sockets in a synchronous way. I.e., the data gets replicated, then the request return. This is the same as the 'synchronous' setting except it uses a pool of sockets, hence it is multithreaded. This is the fastest and safest configuration. To use this, also increase the nr of tcp threads that you have dealing with replication.

* Synchronous means that the thread that executes the request, is also the thread that replicates the data to the other nodes, and will not return until all nodes have received the information.

* Asynchronous means that there is a specific 'sender' thread for each cluster node, so the request thread will queue the replication request into a "smart" queue, and then return to the client.

The "smart" queue is a queue where when a session is added to the queue, and the same session already exists in the queue from a previous request, that session will be replaced in the queue instead of replicating two requests. This almost never happens, unless there is a large network delay.

-->

<!--

When configuring for clustering, you also add in a valve to catch all the requests coming in, at the end of the request, the session may or may not be replicated.

A session is replicated if and only if all the conditions are met:

- AND
1. useDirtyFlag is true or setAttribute or removeAttribute has been called
 2. a session exists (has been created)
 3. the request is not trapped by the "filter" attribute

The filter attribute is to filter out requests that could not modify the session,

hence we don't replicate the session after the end of this request.

The filter is negative, i.e., anything you put in the filter, you mean to filter out,

i.e., no replication will be done on requests that match one of the filters.

The filter attribute is delimited by ;, so you can't escape out ; even if you wanted to.

filter=".*\gif;.*\js;" means that we will not replicate the session after requests with the URI

ending with .gif and .js are intercepted.

The deployer element can be used to deploy apps cluster wide.

Currently the deployment only deploys/undeploys to working members in the cluster

so no WARs are copied upon startup of a broken node.

The deployer watches a directory (watchDir) for WAR files when watchEnabled="true"

When a new war file is added the war gets deployed to the local instance,

and then deployed to the other instances in the cluster.

When a war file is deleted from the watchDir the war is undeployed locally

and cluster wide

-->

<!--

```
<Cluster className="org.apache.catalina.cluster.tcp.SimpleTcpCluster"
```

```
managerClassName="org.apache.catalina.cluster.session.DeltaManager"
```

```
expireSessionsOnShutdown="false"
```

```
useDirtyFlag="true"
```

```
notifyListenersOnReplication="true">
```

```
<Membership
```

```
className="org.apache.catalina.cluster.mcast.McastService"
```

```
mcastAddr="228.0.0.4"
```

```
mcastPort="45564"
```

```
mcastFrequency="500"
```

```
mcastDropTime="3000"/>
```

```
<Receiver
```

```
className="org.apache.catalina.cluster.tcp.ReplicationListener"
```

```
tcpListenAddress="auto"
```

```
tcpListenPort="4001"
```

```
tcpSelectorTimeout="100"
```

```
tcpThreadCount="6"/>
```

```
<Sender
```

```
className="org.apache.catalina.cluster.tcp.ReplicationTransmitter"
```

```
replicationMode="pooled"
```

```
ackTimeout="15000"/>
```

```
<Valve className="org.apache.catalina.cluster.tcp.ReplicationValve"
```

```
filter=".*\.gif;.*\js;.*\jpg;.*\png;.*\htm;.*\html;.*\css;.*\txt;"/>
```

```

    <Deployer
      className="org.apache.catalina.cluster.deploy.FarmWarDeployer"
        tempDir="/tmp/war-temp/"
        deployDir="/tmp/war-deploy/"
        watchDir="/tmp/war-listen/"
        watchEnabled="false"/>
    </Cluster>
-->

```

<!-- Normally, users must authenticate themselves to each web app individually. Uncomment the following entry if you would like a user to be authenticated the first time they encounter a resource protected by a security constraint, and then have that user identity maintained across *all* web applications contained in this virtual host. -->

```

<!--
<Valve className="org.apache.catalina.authenticator.SingleSignOn" />
-->

```

<!-- Access log processes all requests for this virtual host. By default, log files are created in the "logs" directory relative to \$CATALINA_HOME. If you wish, you can specify a different directory with the "directory" attribute. Specify either a relative (to \$CATALINA_HOME) or absolute path to the desired directory.

```

-->
<!--
<Valve className="org.apache.catalina.valves.AccessLogValve"
  directory="logs" prefix="localhost_access_log." suffix=".txt"
  pattern="common" resolveHosts="false"/>
-->

```

<!-- Access log processes all requests for this virtual host. By default, log files are created in the "logs" directory relative to \$CATALINA_HOME. If you wish, you can specify a different directory with the "directory" attribute. Specify either a relative (to \$CATALINA_HOME) or absolute path to the desired directory. This access log implementation is optimized for maximum performance, but is hardcoded to support only the "common" and "combined" patterns.

```

-->
<!--
<Valve
  className="org.apache.catalina.valves.FastCommonAccessLogValve"
    directory="logs" prefix="localhost_access_log." suffix=".txt"

```

```

        pattern="common" resolveHosts="false"/>
-->
<!-- Access log processes all requests for this virtual host. By
default, log files are created in the "logs" directory relative to
$CATALINA_HOME. If you wish, you can specify a different
directory with the "directory" attribute. Specify either a relative
(to $CATALINA_HOME) or absolute path to the desired directory.
This access log implementation is optimized for maximum performance,
but is hardcoded to support only the "common" and "combined"
patterns.

```

This valve use NIO direct Byte Buffer to asynchronously store the log.

```

-->
<!--
<Valve
className="org.apache.catalina.valves.ByteBufferAccessLogValve"
    directory="logs" prefix="localhost_access_log." suffix=".txt"
    pattern="common" resolveHosts="false"/>
-->
<Context path="/sinu-zkoss" docBase="C:\PROYECTOS\SINU\build\sinu-
zkoss\wars\sinu-zkoss.war" debug="0" reloadable="true" crossContext="true"
privileged="true">
    <Logger className="org.apache.catalina.logger.FileLogger" directory="logs"
prefix="localhost_sinu_log." suffix=".txt" timestamp="true"/>
    <Resource name="jdbc/sinuDS" auth="Container"
        type="javax.sql.DataSource"
        username="sinu"
        password="sinu"
        driverClassName="oracle.jdbc.driver.OracleDriver"
        url="jdbc:oracle:thin:@192.168.10.21:1521:oraclep"
        maxWait="10000"
        maxActive="20"
        maxIdle="10"
        removeAbandoned="true"
        removeAbandonedTimeout="180"/>
</Context>

    <Valve className="org.josso.tc55.agent.SSOAgentValve" debug="1"/>

</Host>

</Engine>

</Service>

```

</Server>

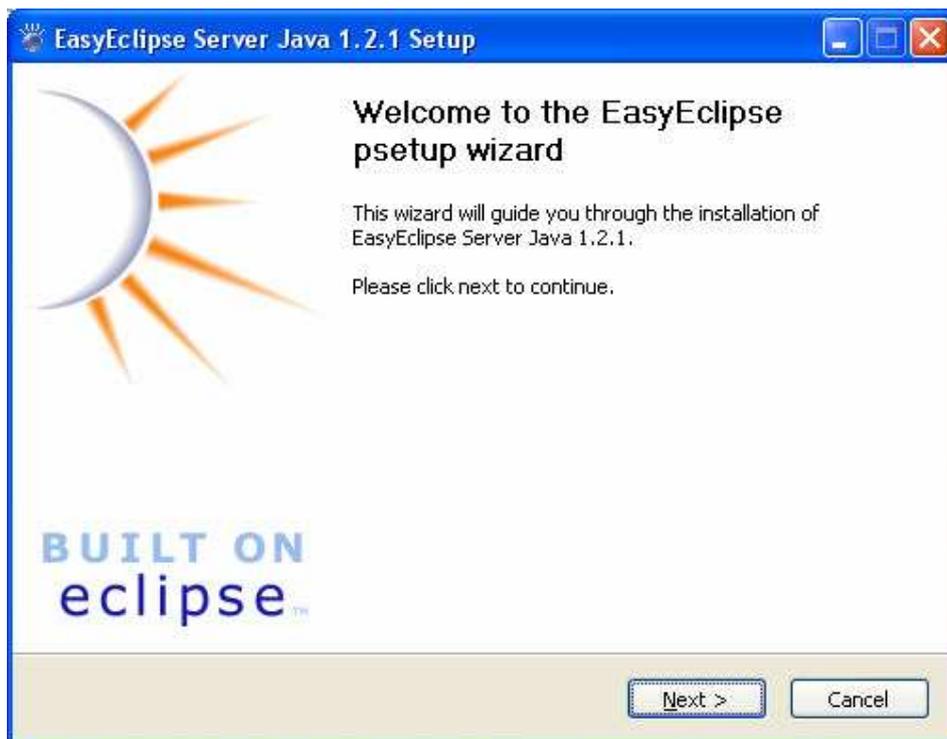
Nota:

El Tomcat que se entrega en el directorio Software, se encuentra configurado para trabajar correctamente por el puerto 8087.

Instalar el IDE

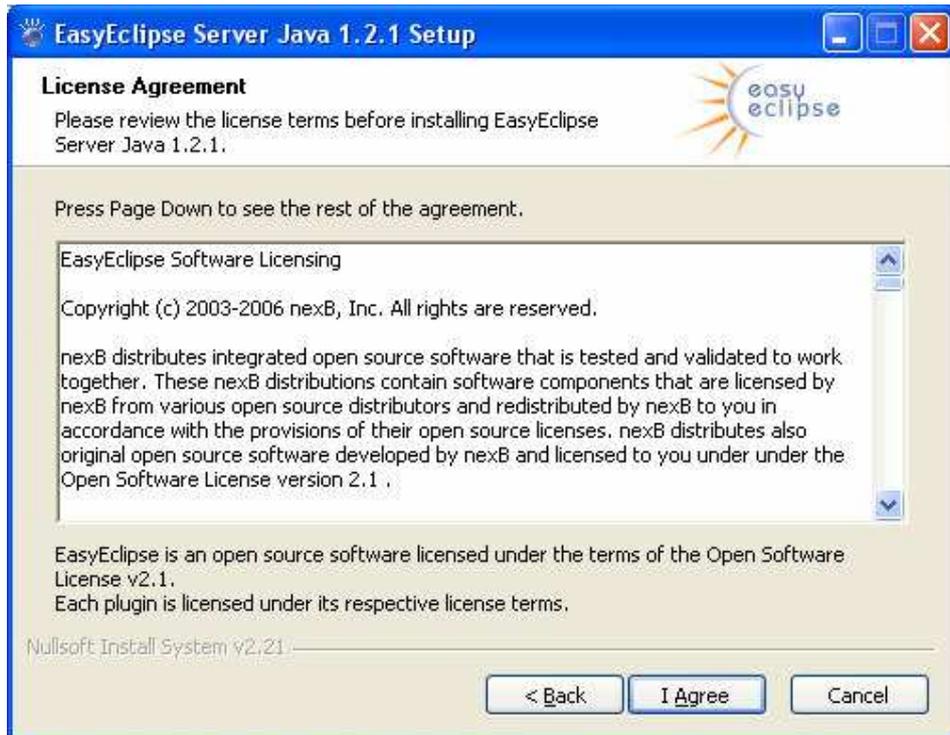
Eclipse

Ejecutar el archivo de instalación de Eclipse

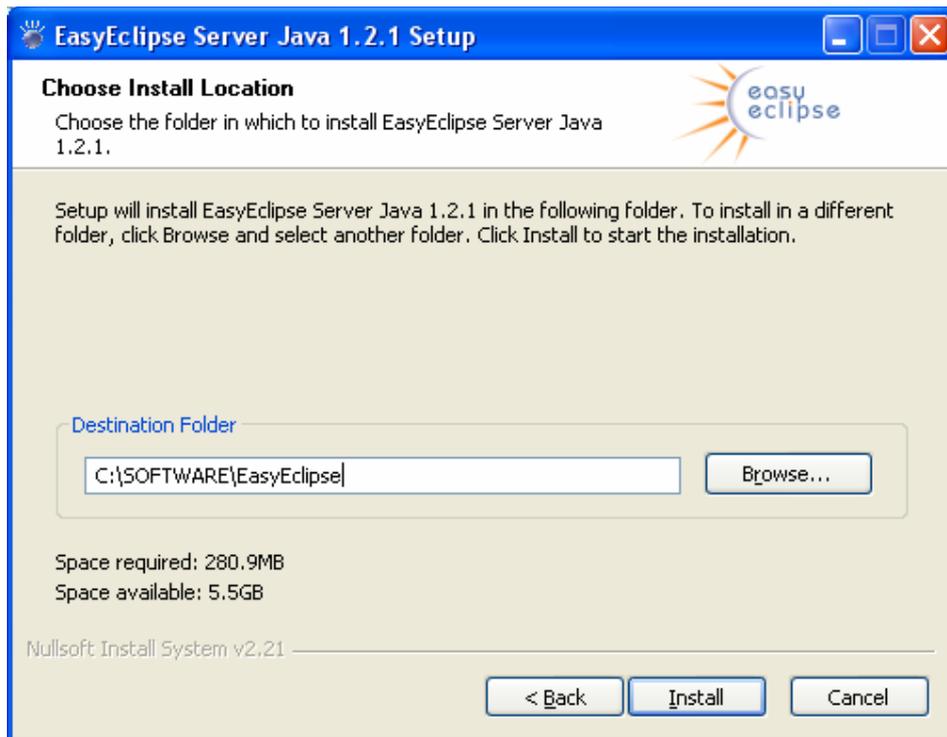


Clic en Next.

Aceptamos la Licencia



Escogemos el directorio de Instalación. (En nuestra carpeta SOFTWARE)



Presionamos Install

Configuración del IDE

**Crear una carpeta llamada SINU dentro del directorio
PROYECTOS**

**Copiar en esta carpeta SINU los siguientes directorios dados
por ACIES:**

- ext
- build
- sinu

Copiamos el proyecto en esta carpeta (SINU).

Ejecutar Eclipse

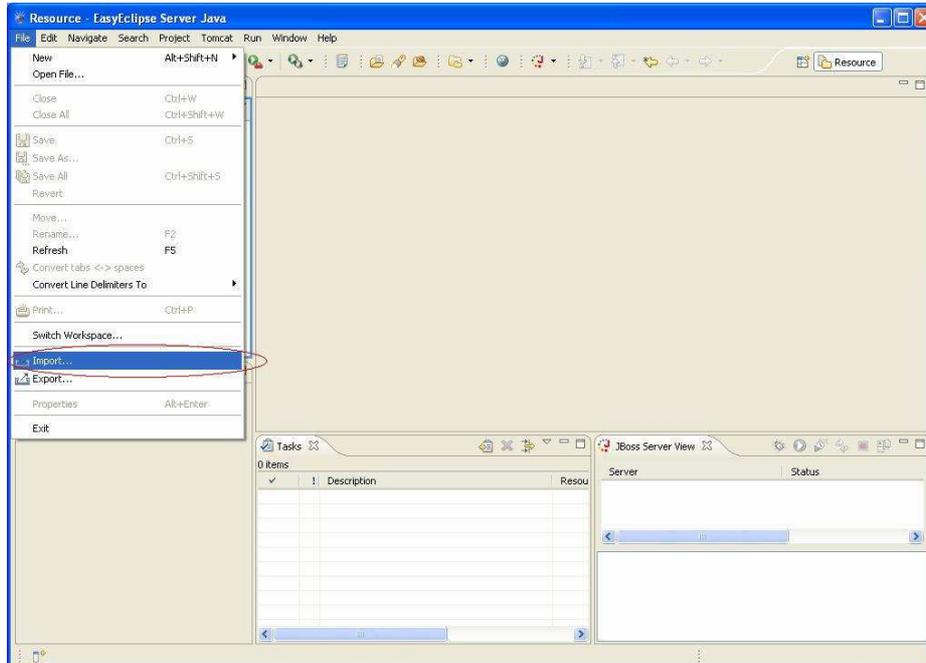


Seleccionar el espacio de Trabajo.

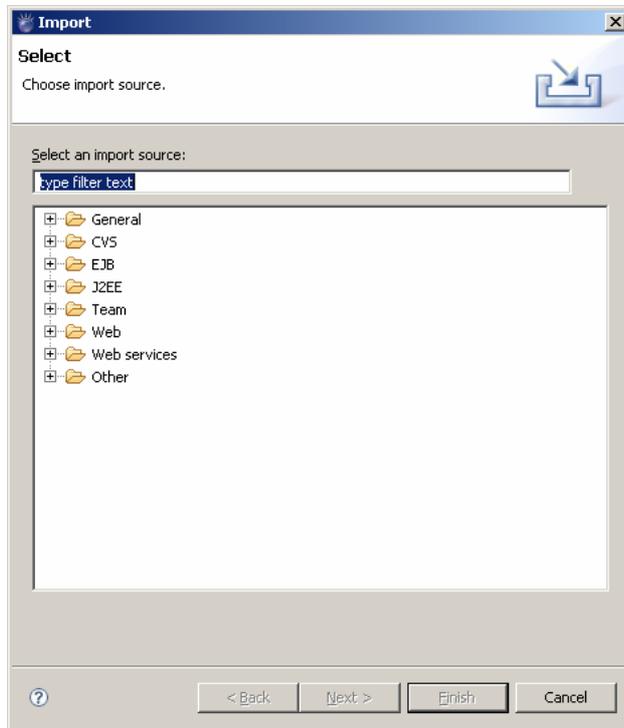


**Damos clic en OK y luego en Next, las veces que sean
necesarias, hasta configurar nuestro espacio de trabajo.**

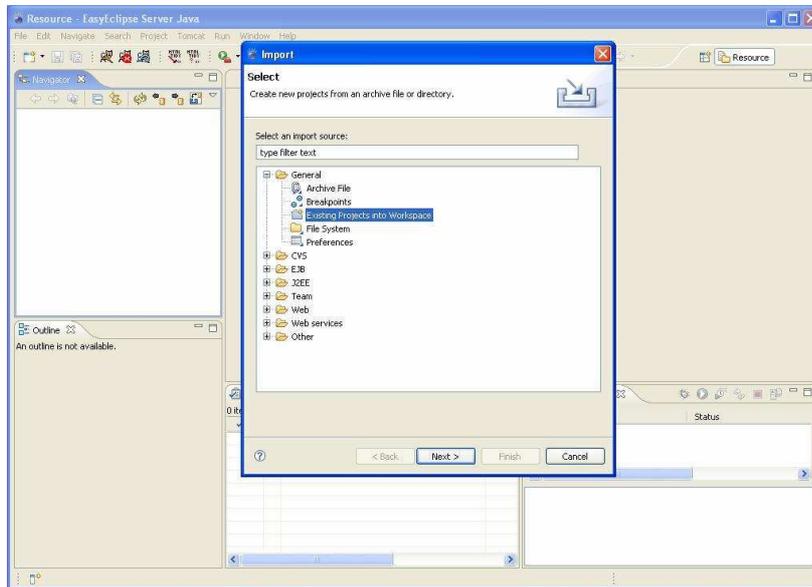
Importar el proyecto



Luego aparece la siguiente pantalla, en la cual vamos a importar nuestro proyecto.

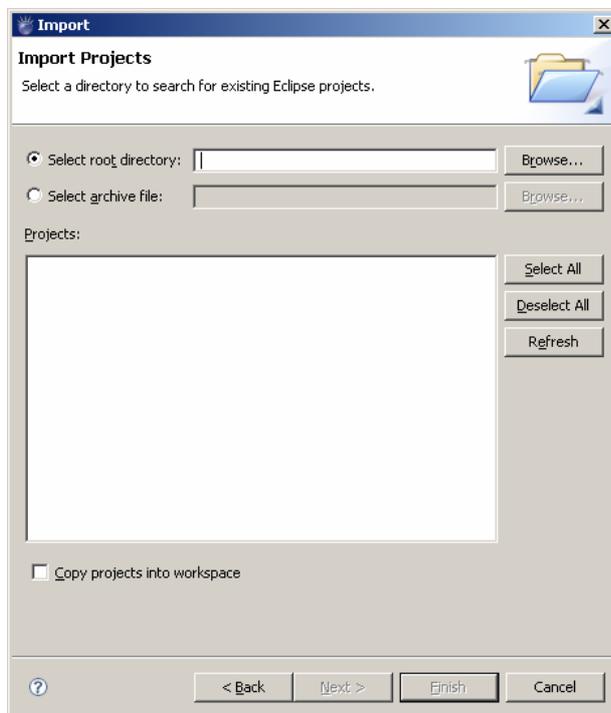


Se debe seleccionar General/Existing Projects into Workspace



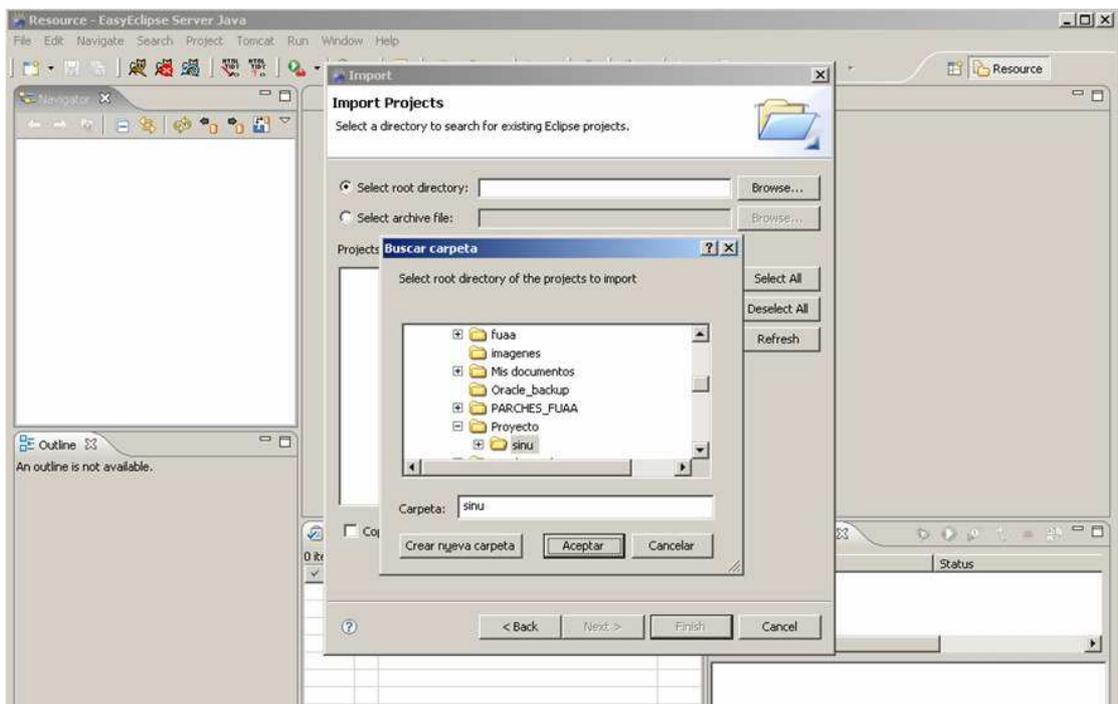
Oprimir en el botón Next

Aparece la siguiente ventana



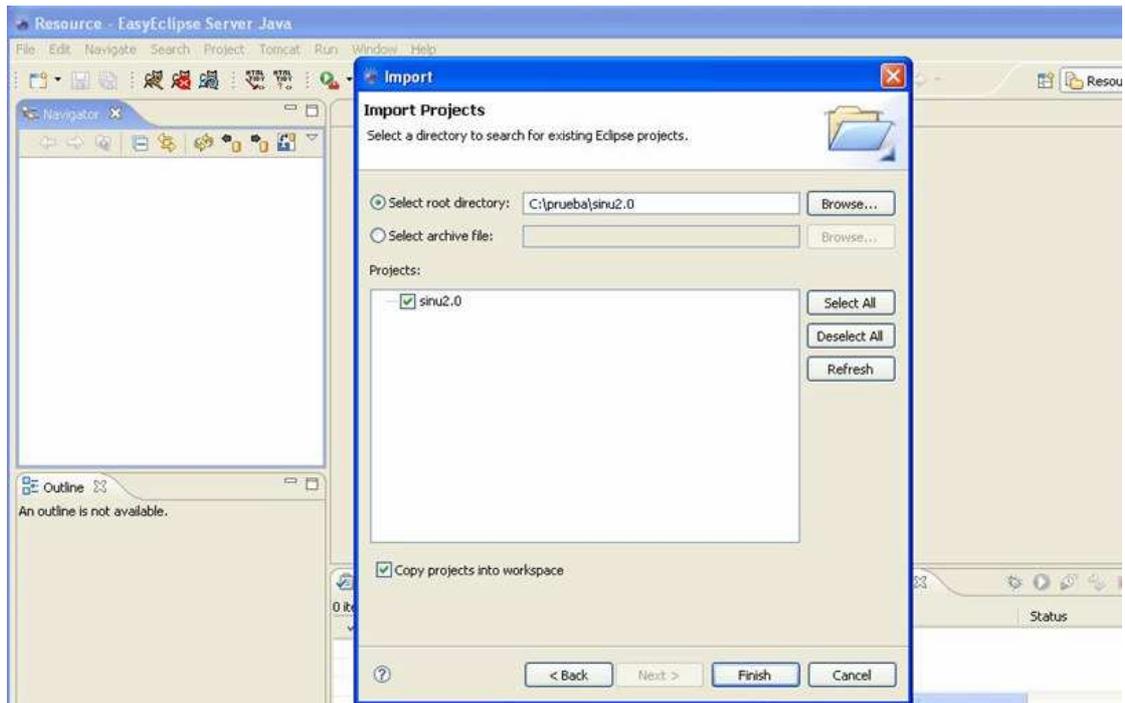
Oprimos en Browse... Para buscar el proyecto a importar.

Buscamos el proyecto

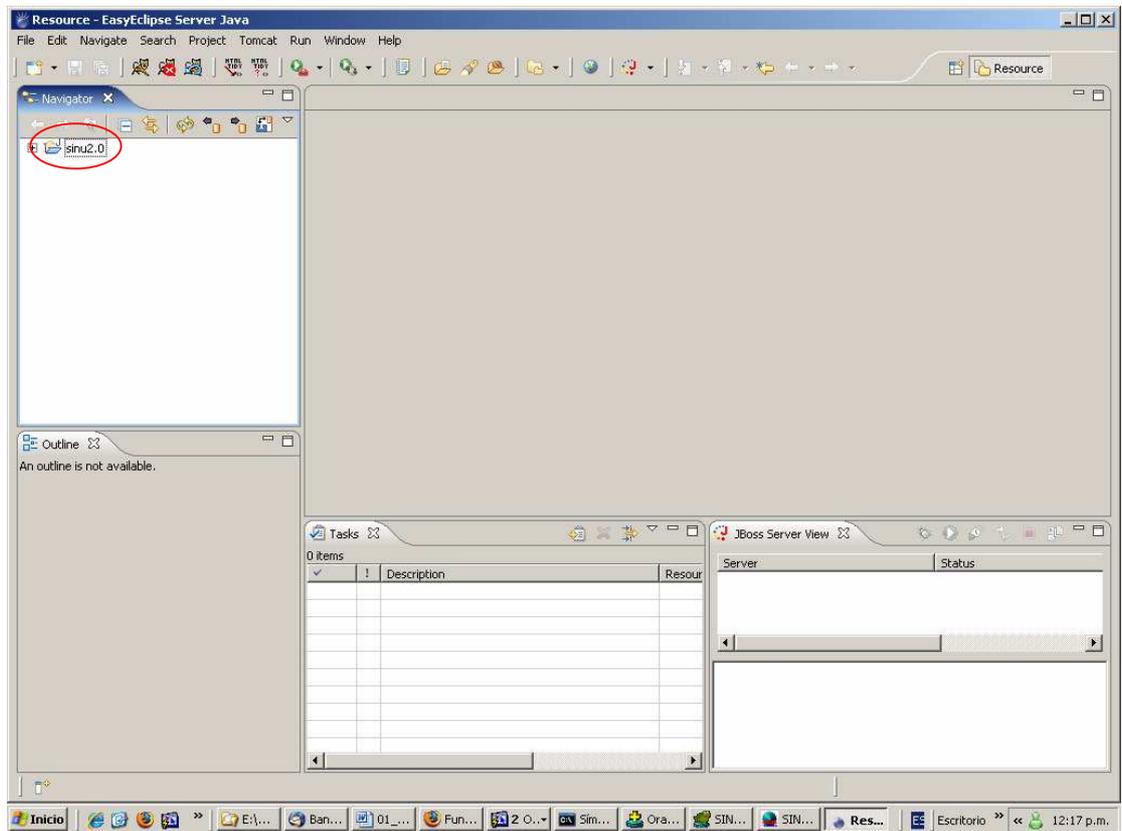


Oprimimos en Aceptar

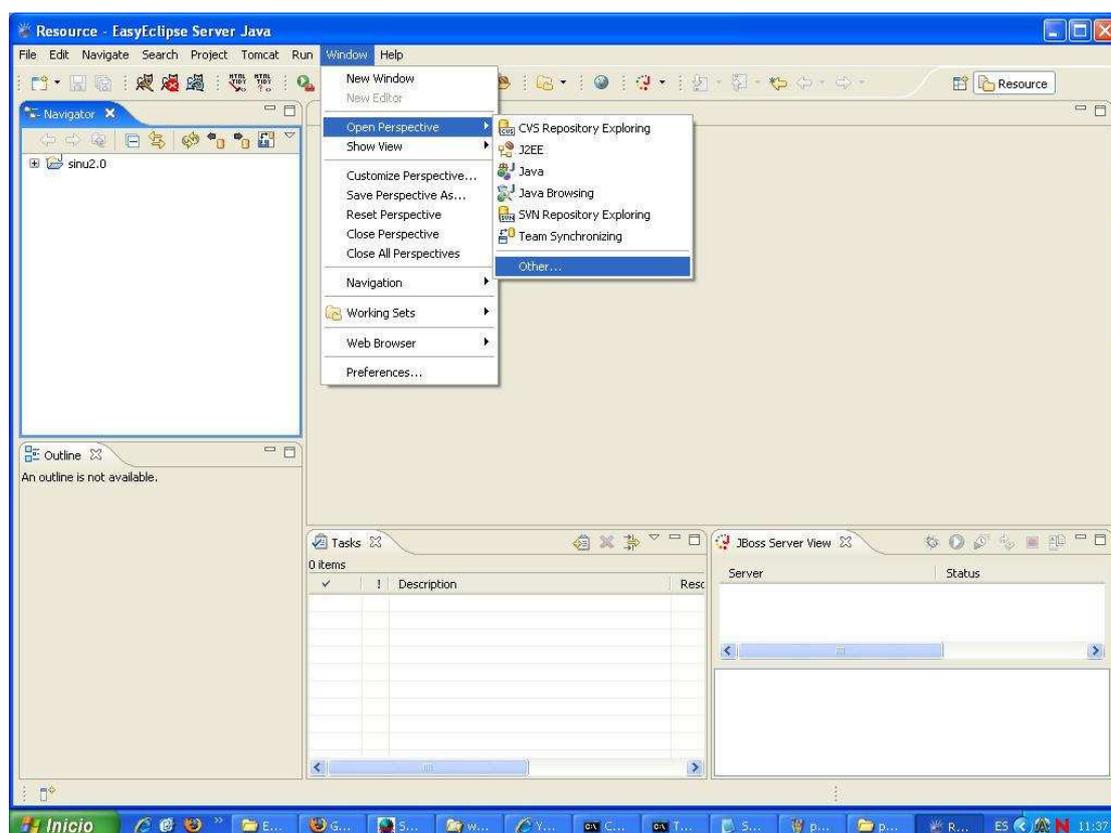
La ventana queda de la siguiente forma y damos clic en Finalizar



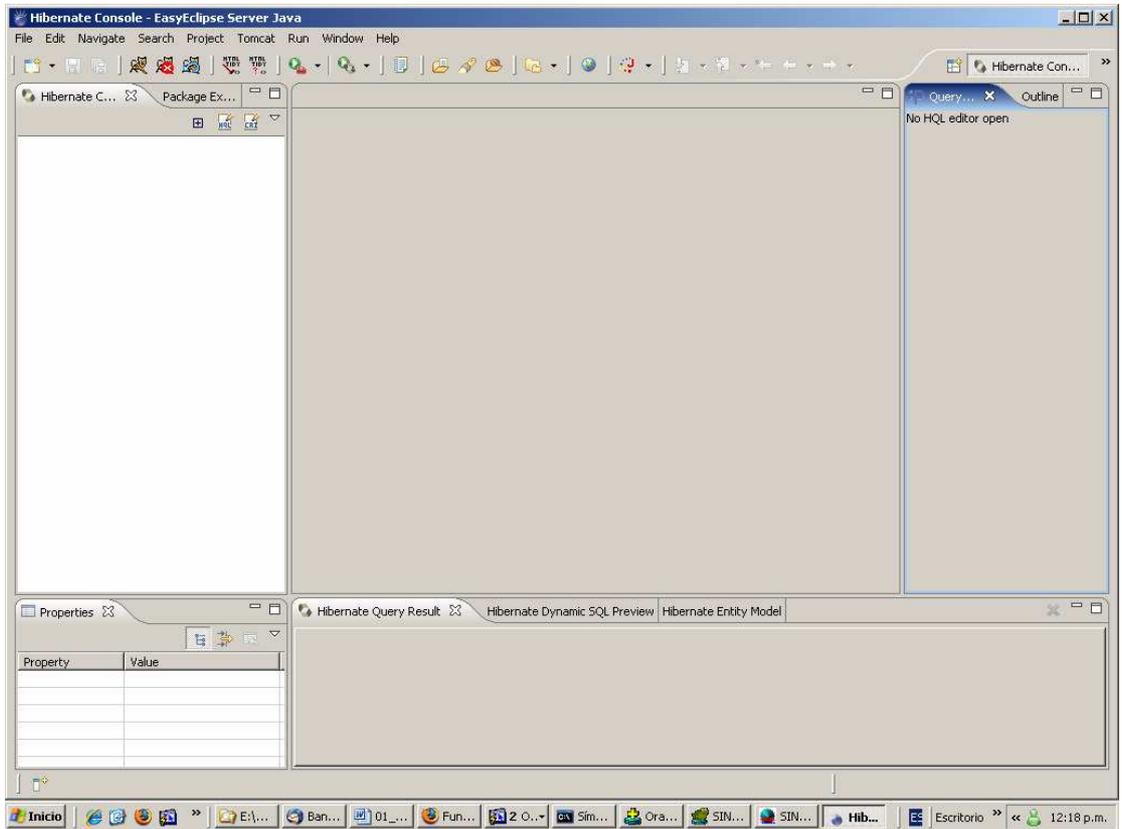
Luego debe aparecer el proyecto (sinu2.0) en la parte superior de la ventana.



Seleccionamos la perspectiva de Hibernate

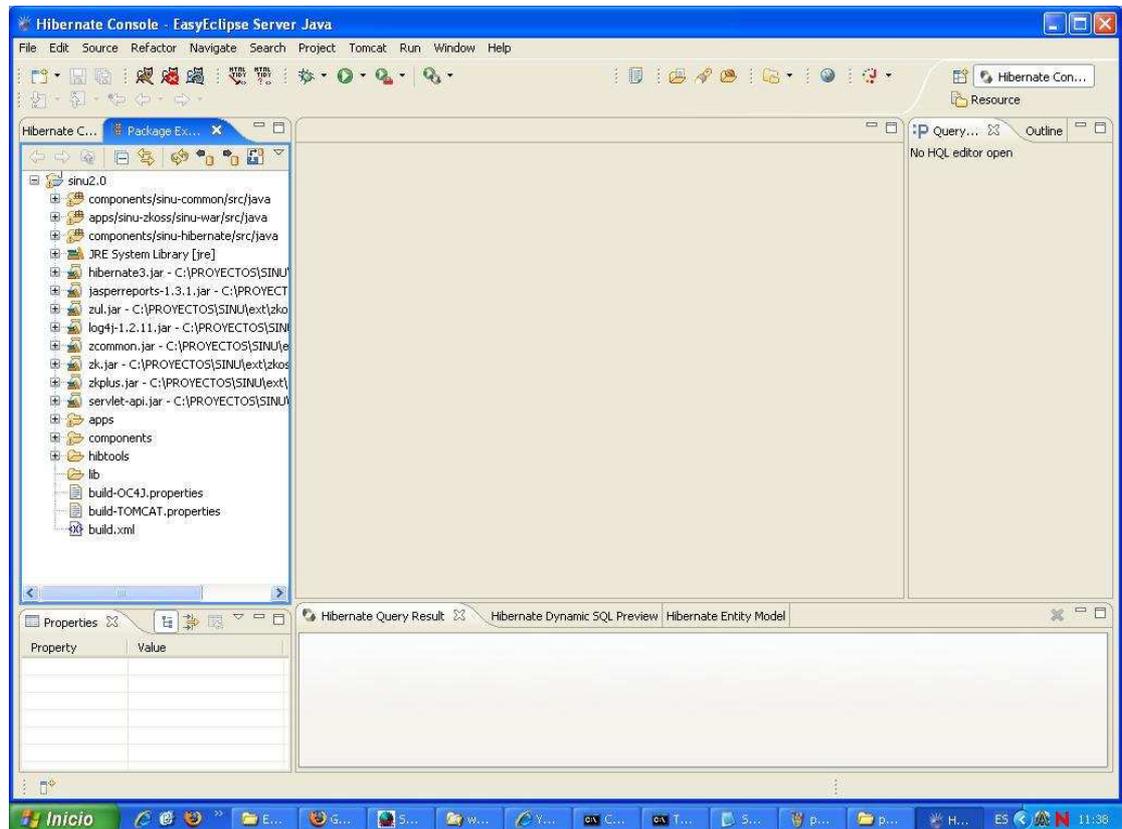


Oprimir el botón OK, y la ventana queda así:



Observamos la perspectiva de Hibernate

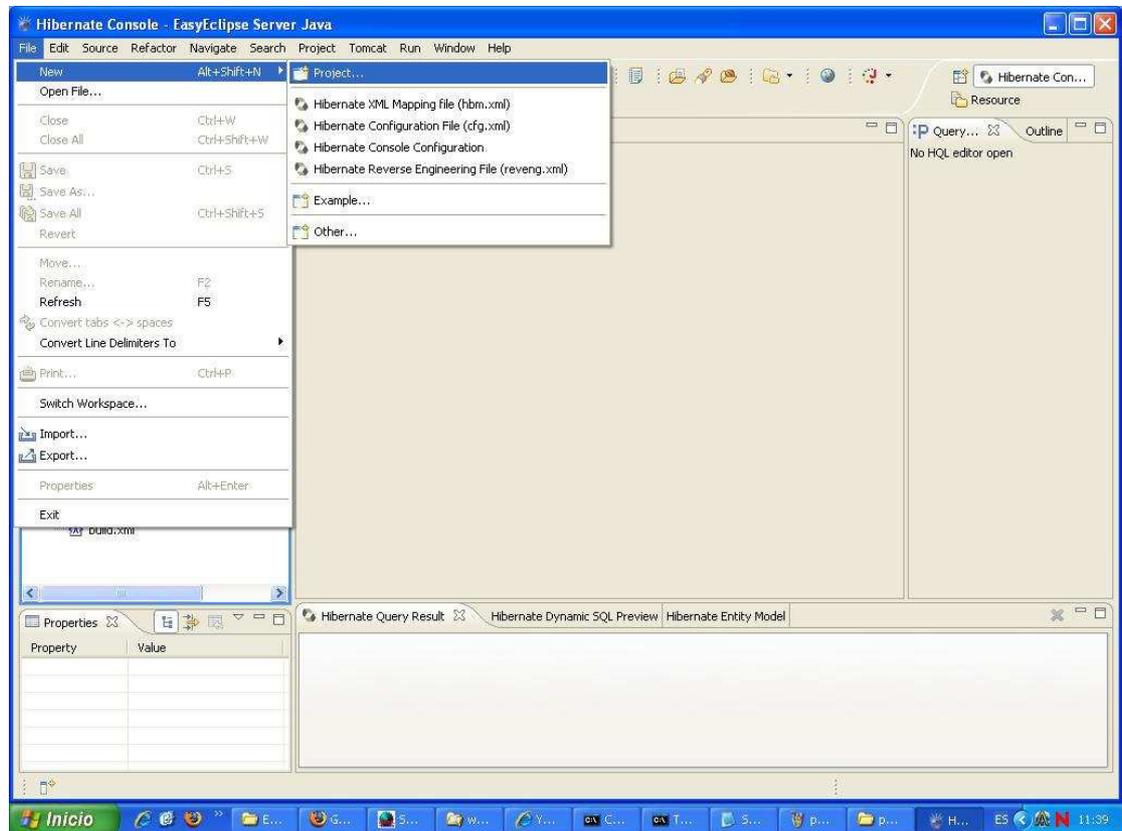
Damos clic en Package Explorer



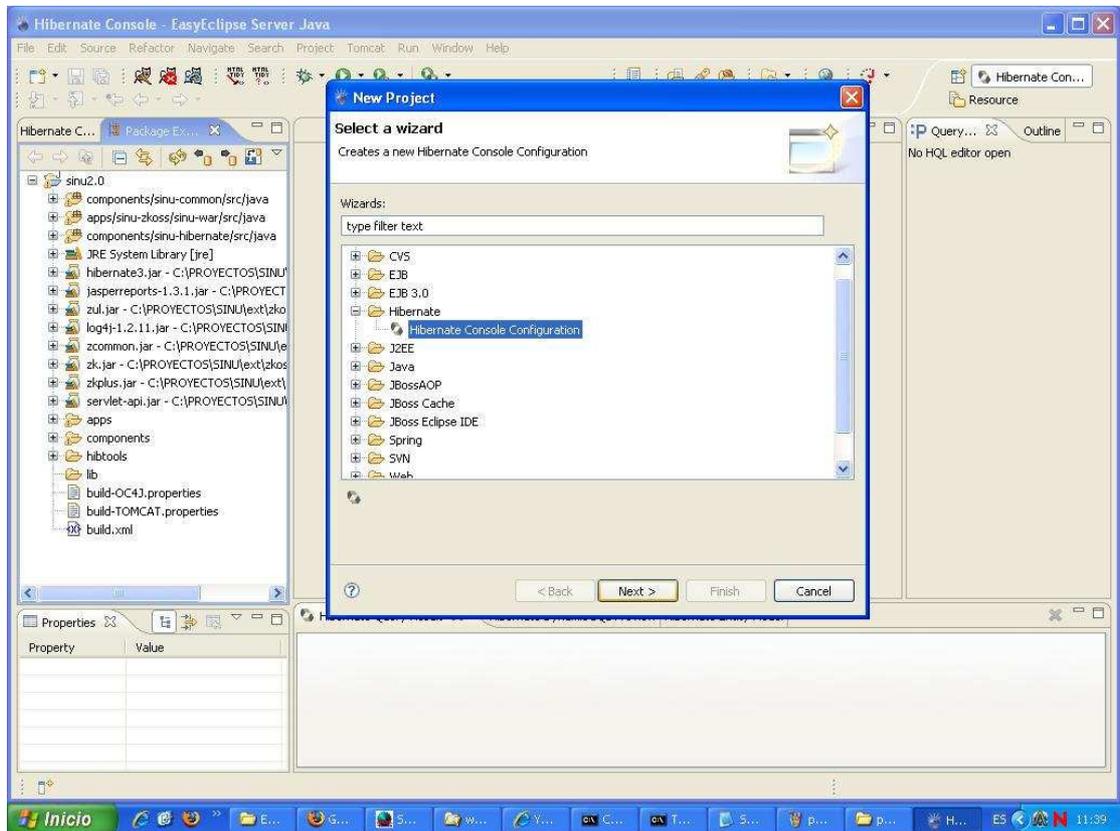
NOTA:

Al importar el proyecto dado por ACIES, y copiar el espacio de trabajo, se pueden obviar los pasos a continuación.

Configuración de la Consola Hibernate



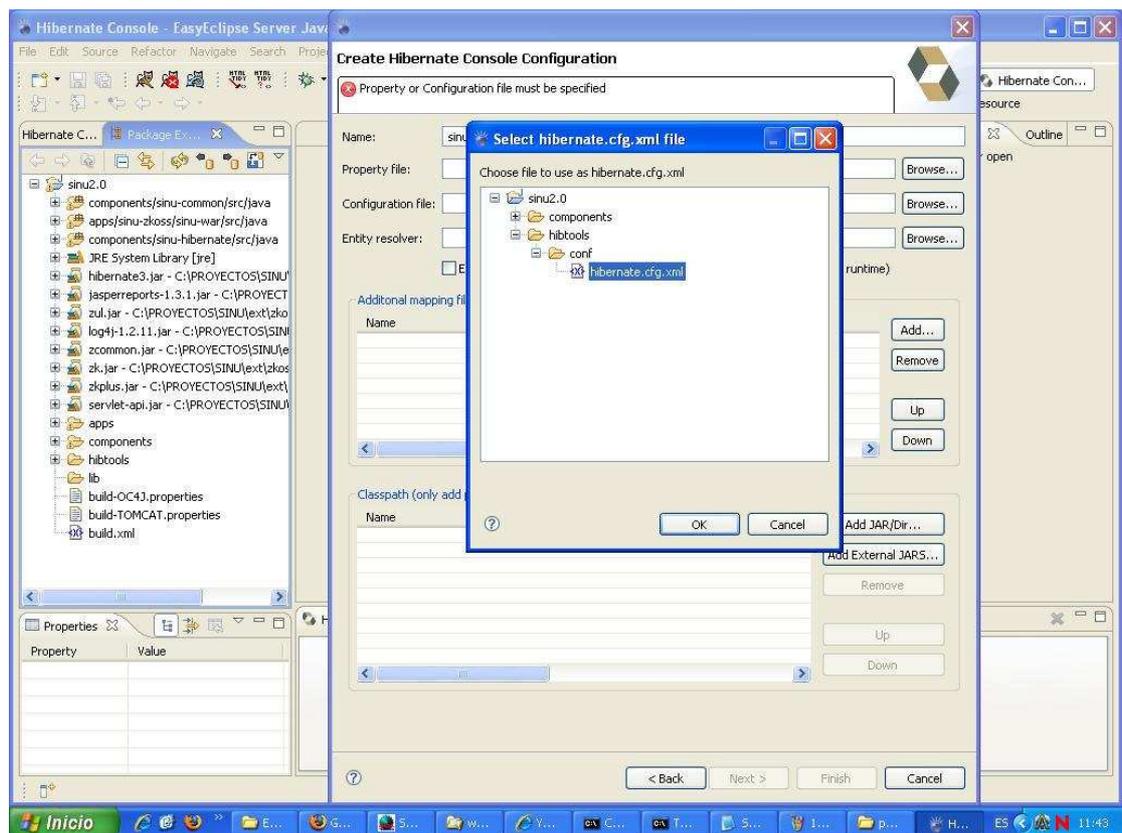
En New Project, seleccionamos Hibernate / Hibernate Console Configuration



Damos clic en Next.

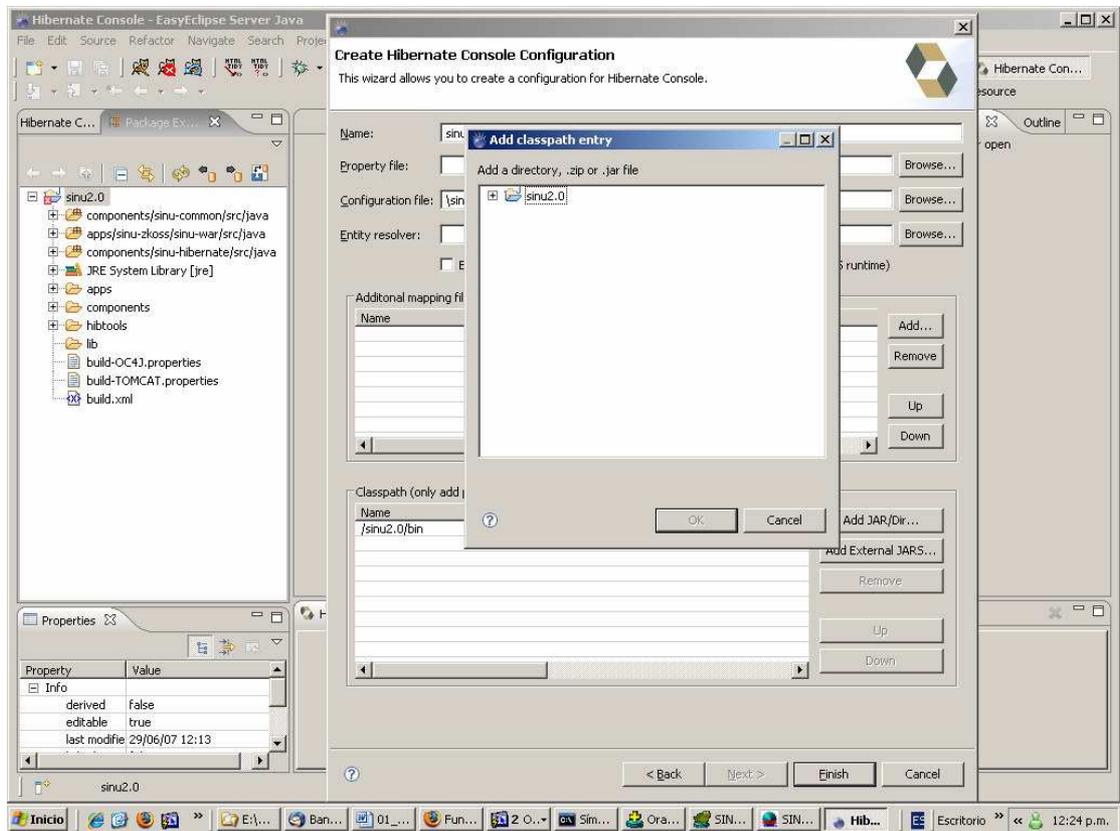
En Name, digitamos: sinu2.0 y continuamos con el siguiente punto sin salirse de esta opción

En configuration file, seleccionamos browse, y luego buscamos hibernate.cfg.xml

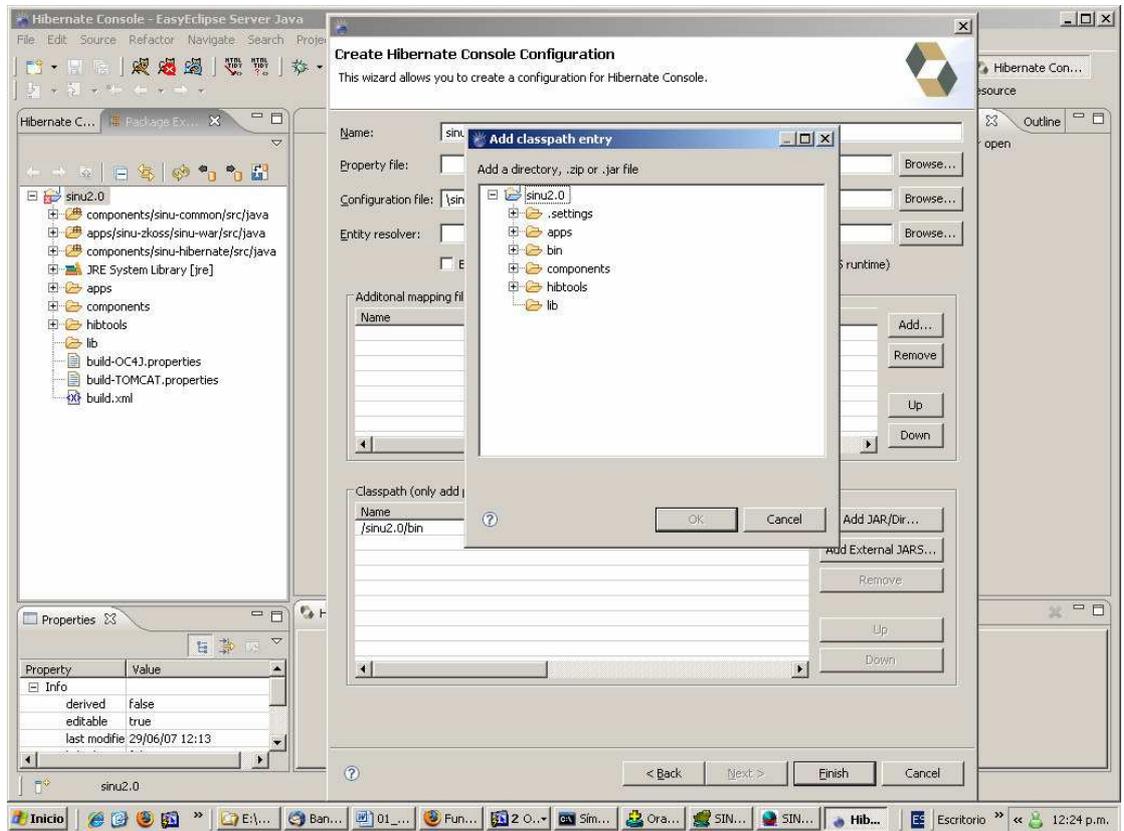


Oprimimos el botón OK

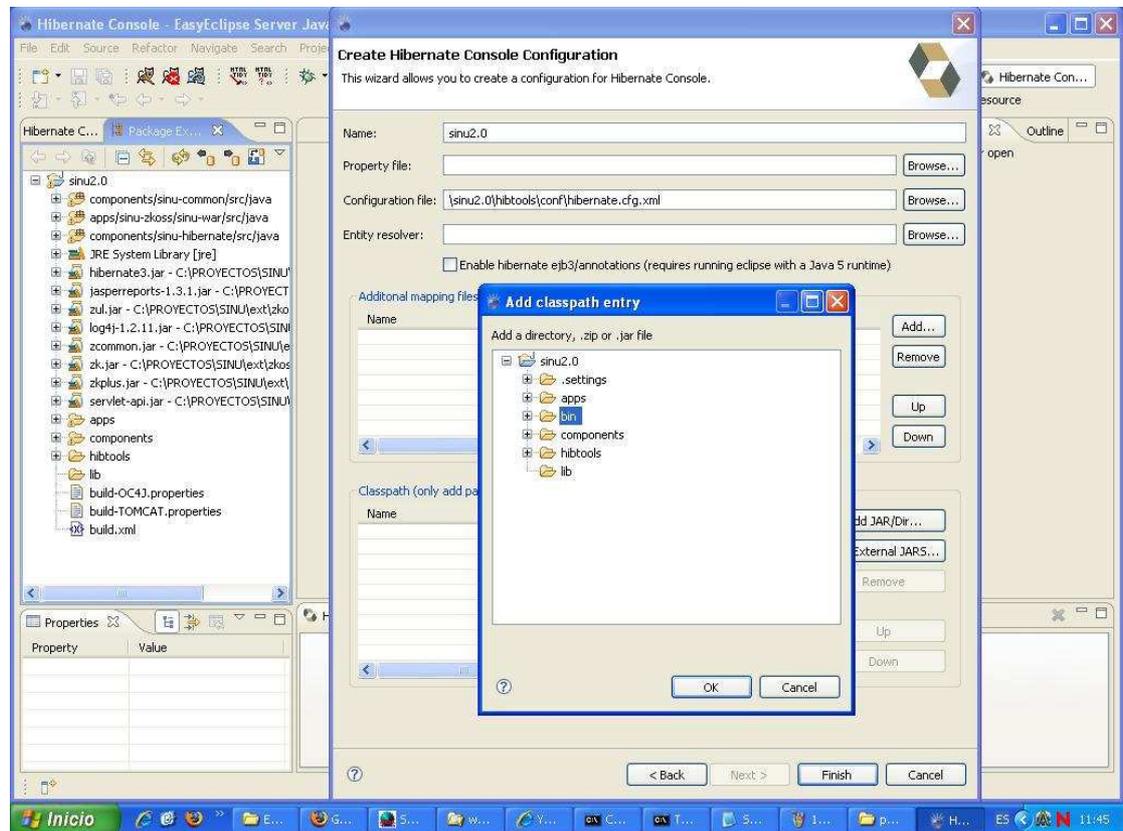
Damos clic en botón Add JAR/DIR,



Luego damos clic en sinu2.0

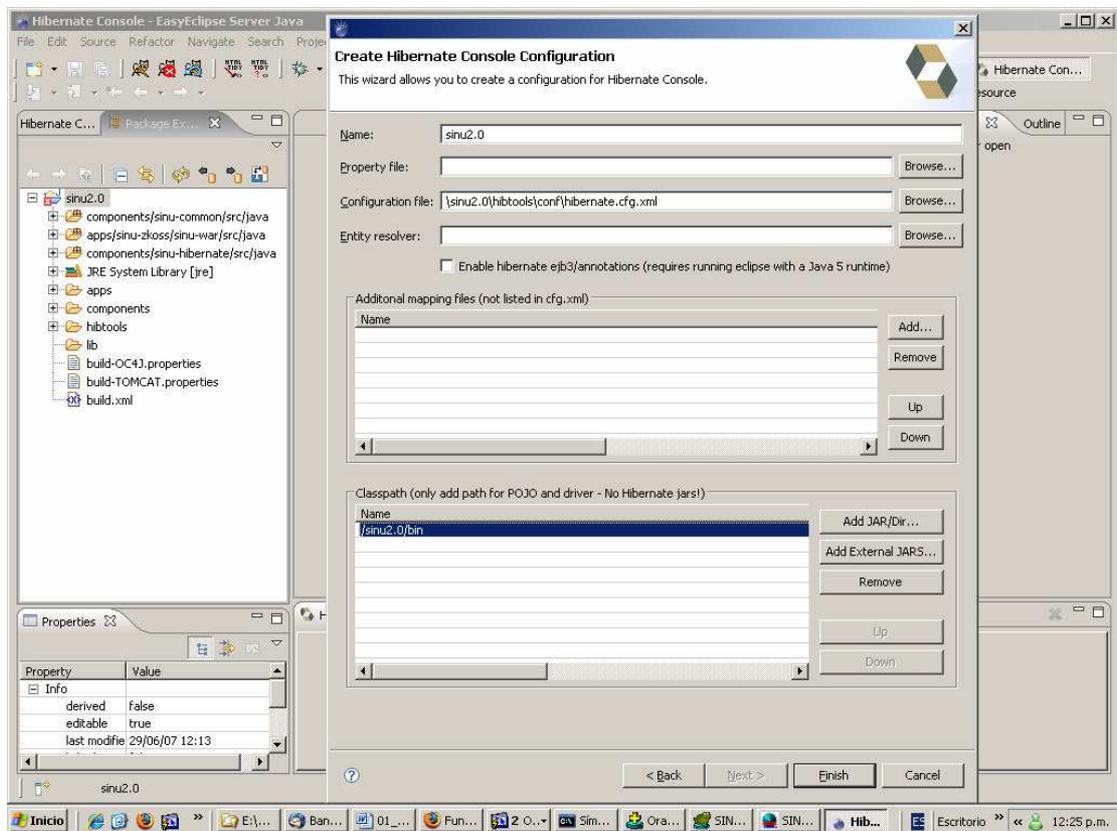


Seleccionamos bin

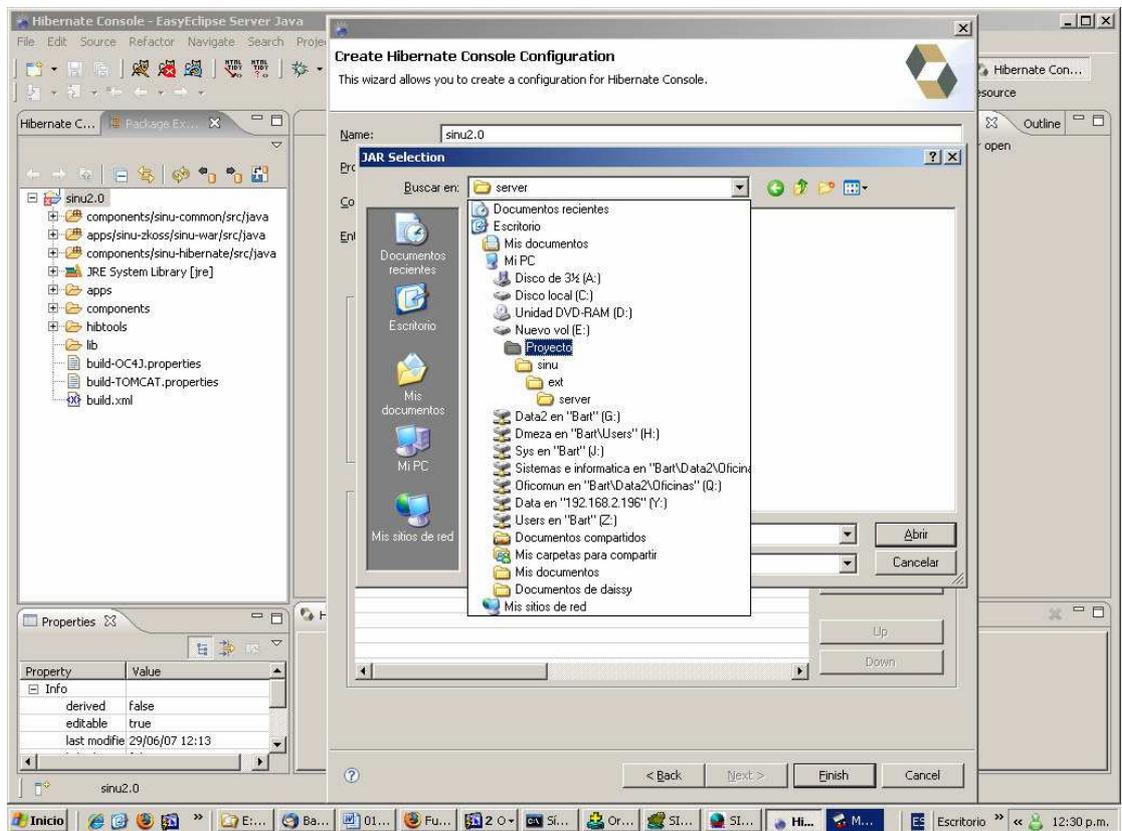


Optimimos OK

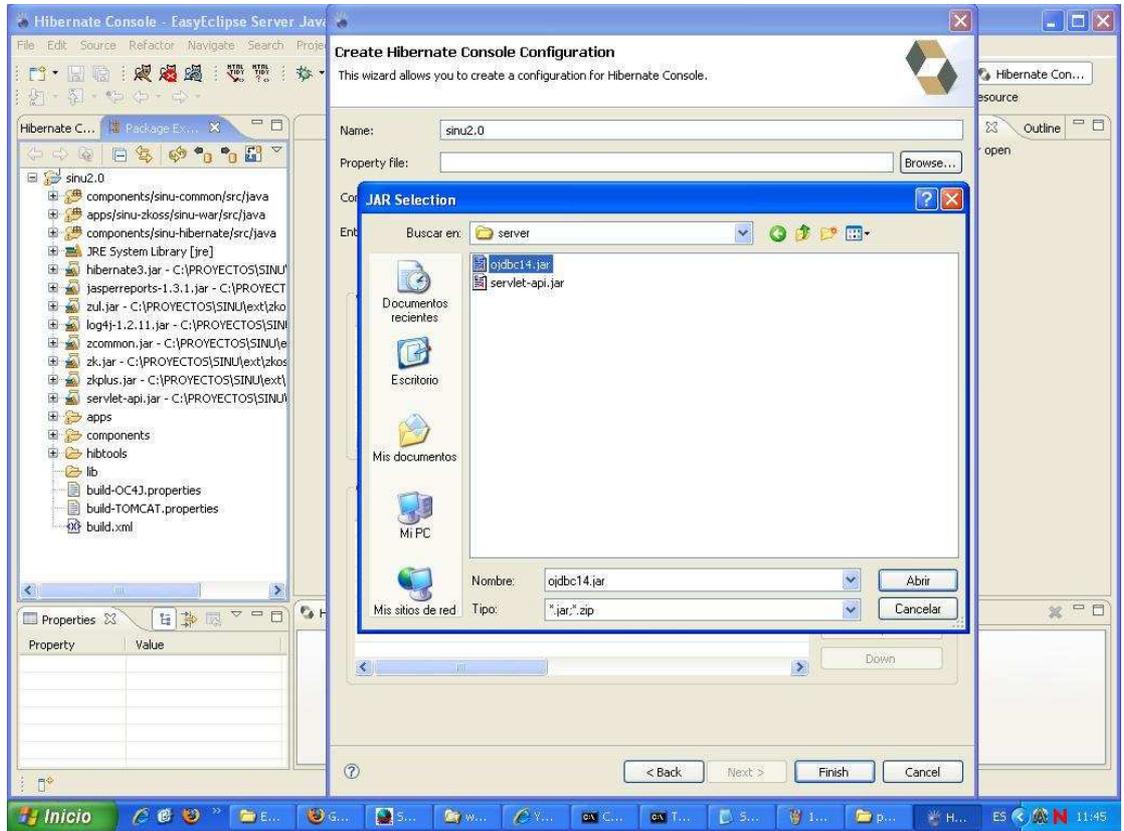
La ventana queda de la siguiente forma:



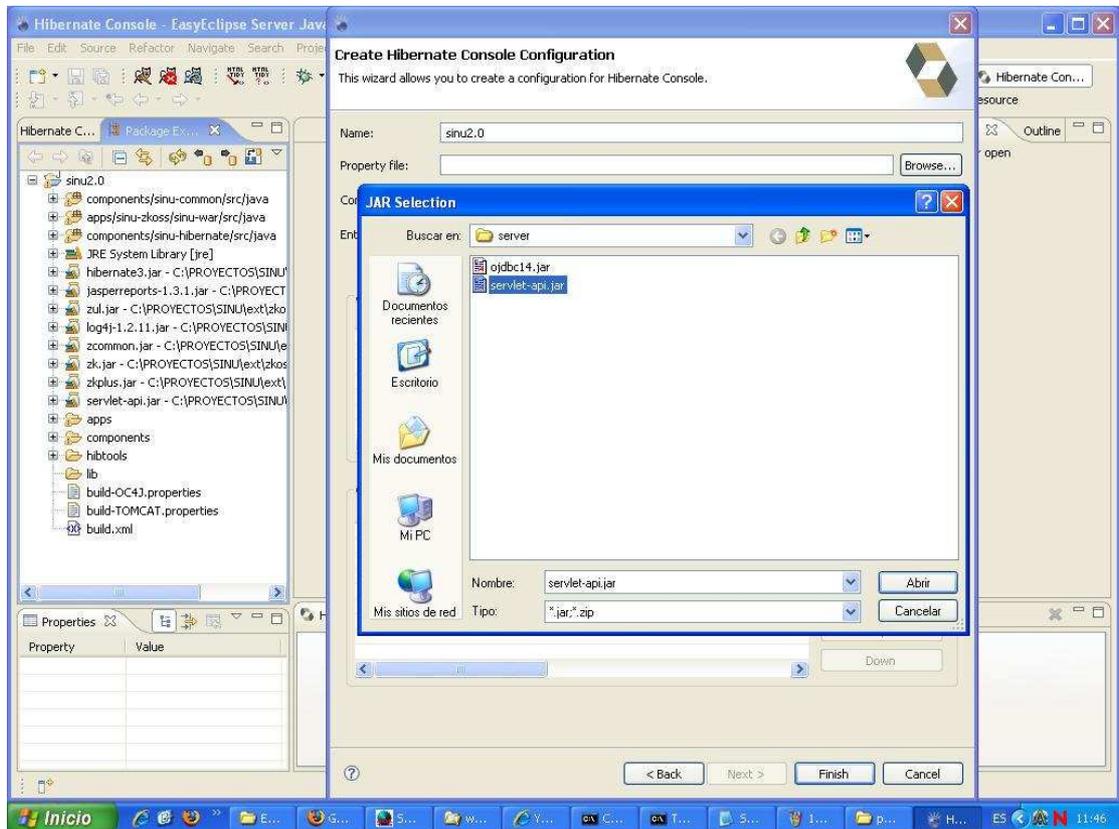
En esta misma ventana, damos clic en External JARS, y seleccionamos ojdbc14.jar y servlet-api.jar (Se encuentran en C:\PROYECTOS\SINU\ext\server)



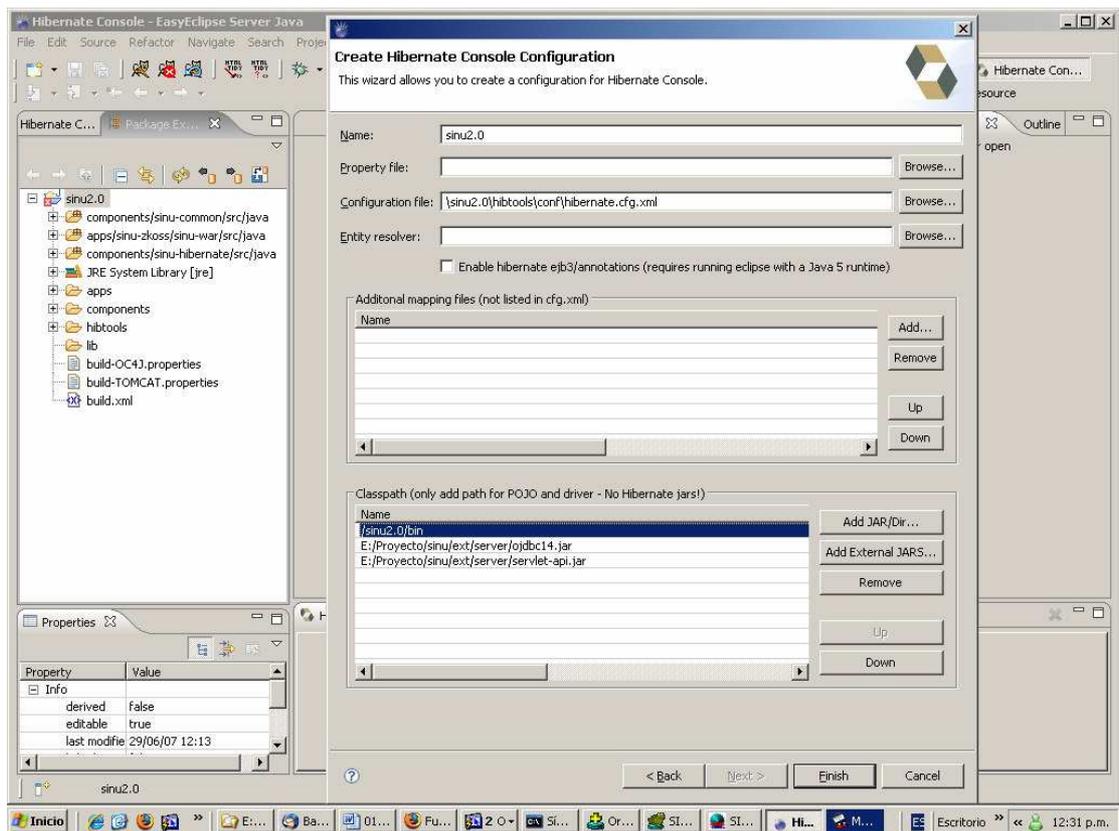
Seleccionamos ojdbc14.jar y damos clic en Abrir



Seleccionamos servlet-api.jar y damos clic en Abrir

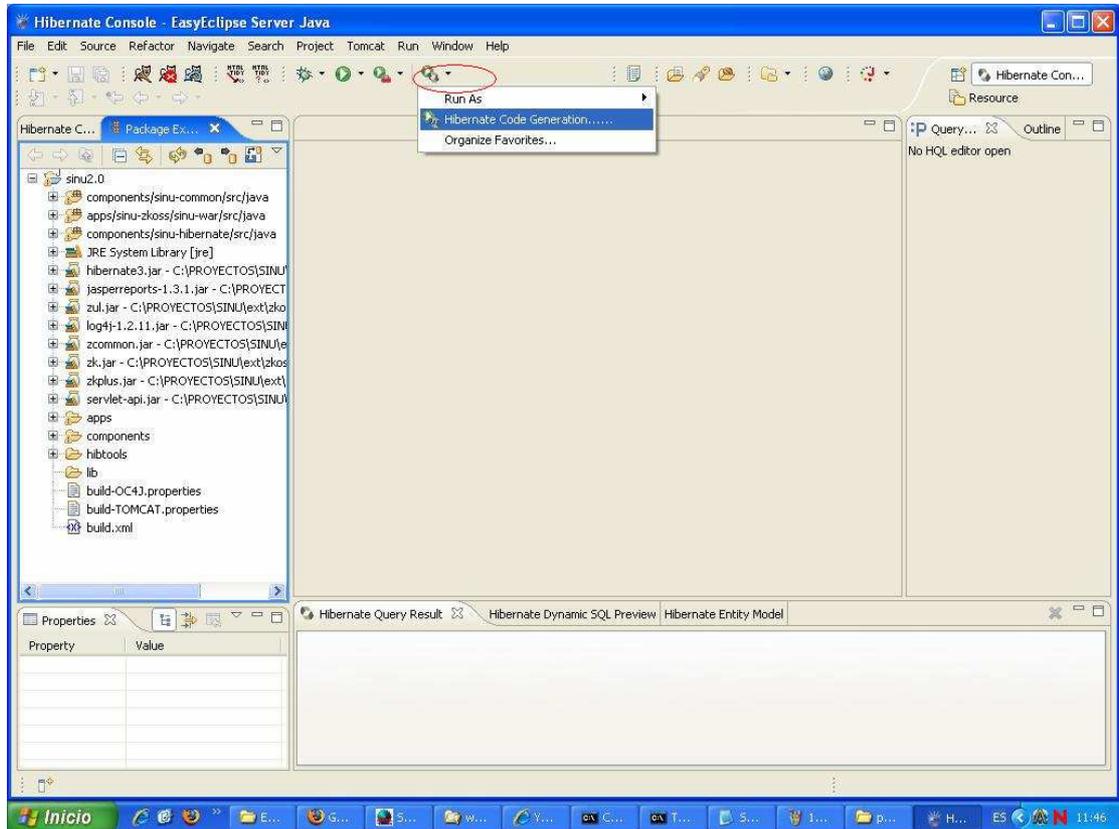


La consola de Configuración de Hibernate queda así:

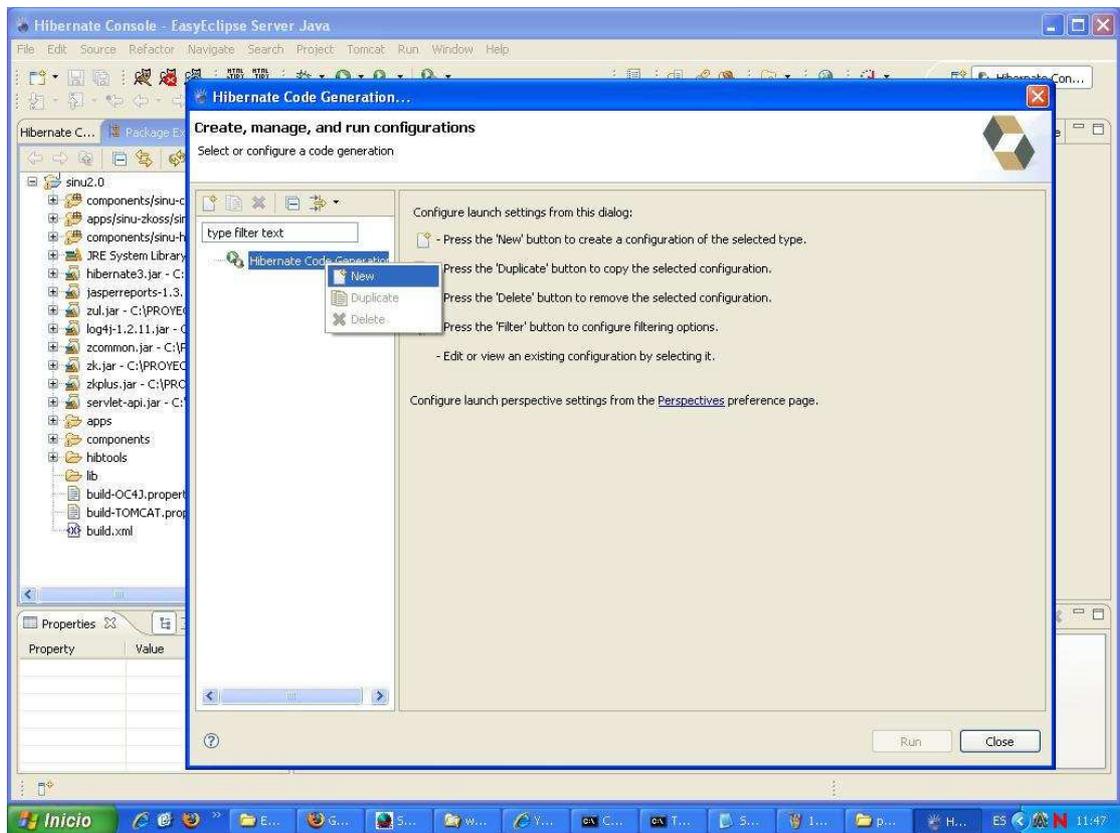


Se oprime el botón Finish

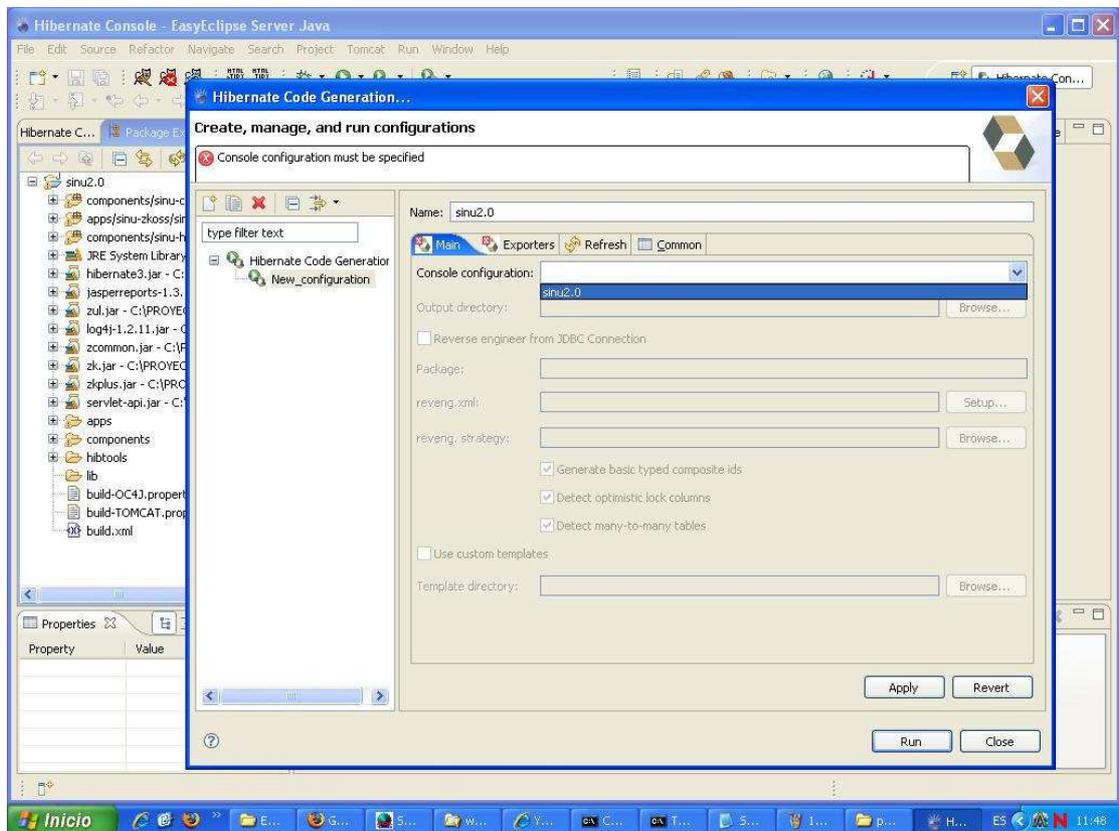
Seleccionamos Hibernate Code Generation...



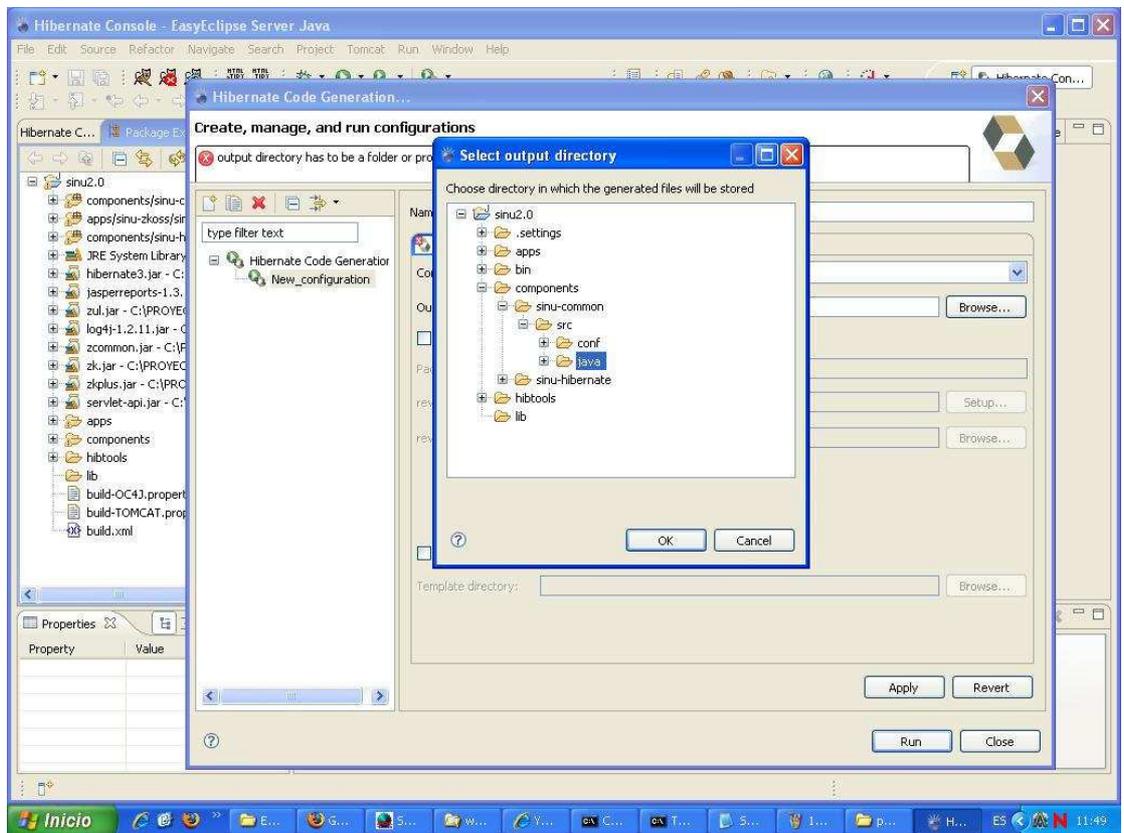
Se despliega la siguiente ventana, y seleccionamos New.



Digitamos en Name: sinu2.0, y lo seleccionamos en Console configuration

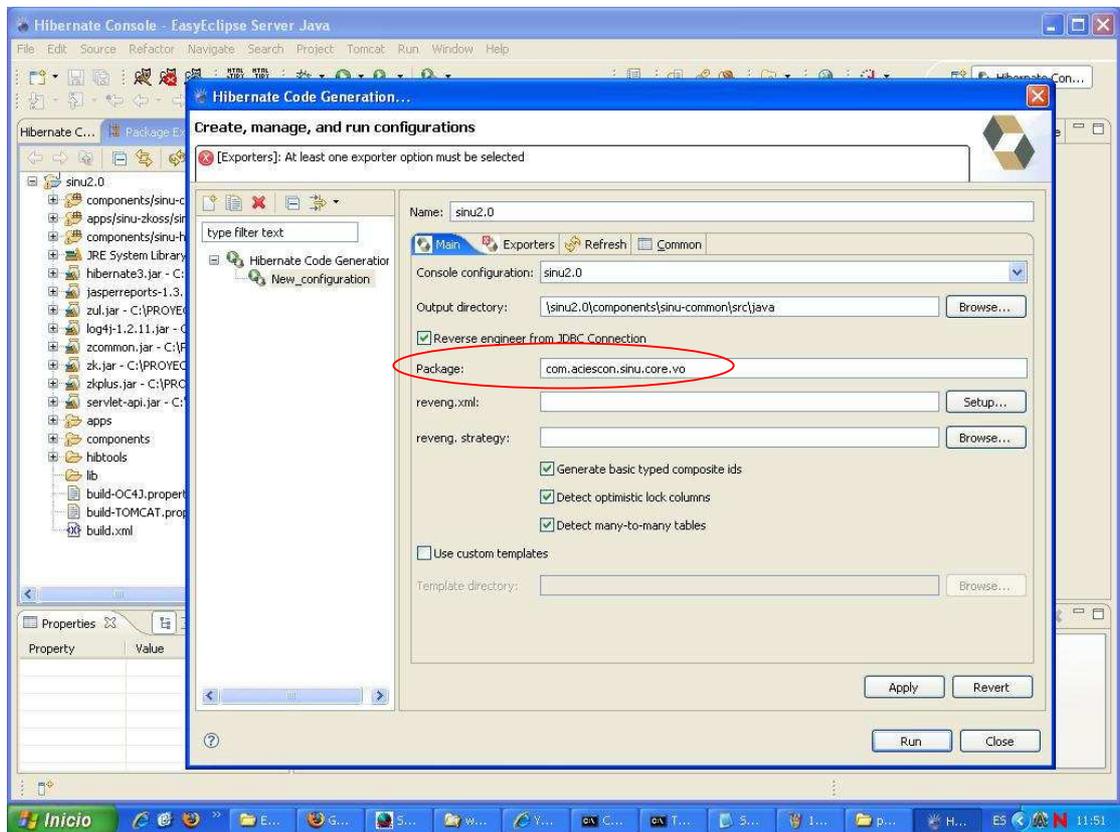


Activamos el check box Reverse engineer from JDBC Connection, y luego seleccionamos el Output Directory seleccionamos java, siguiendo la siguiente ruta

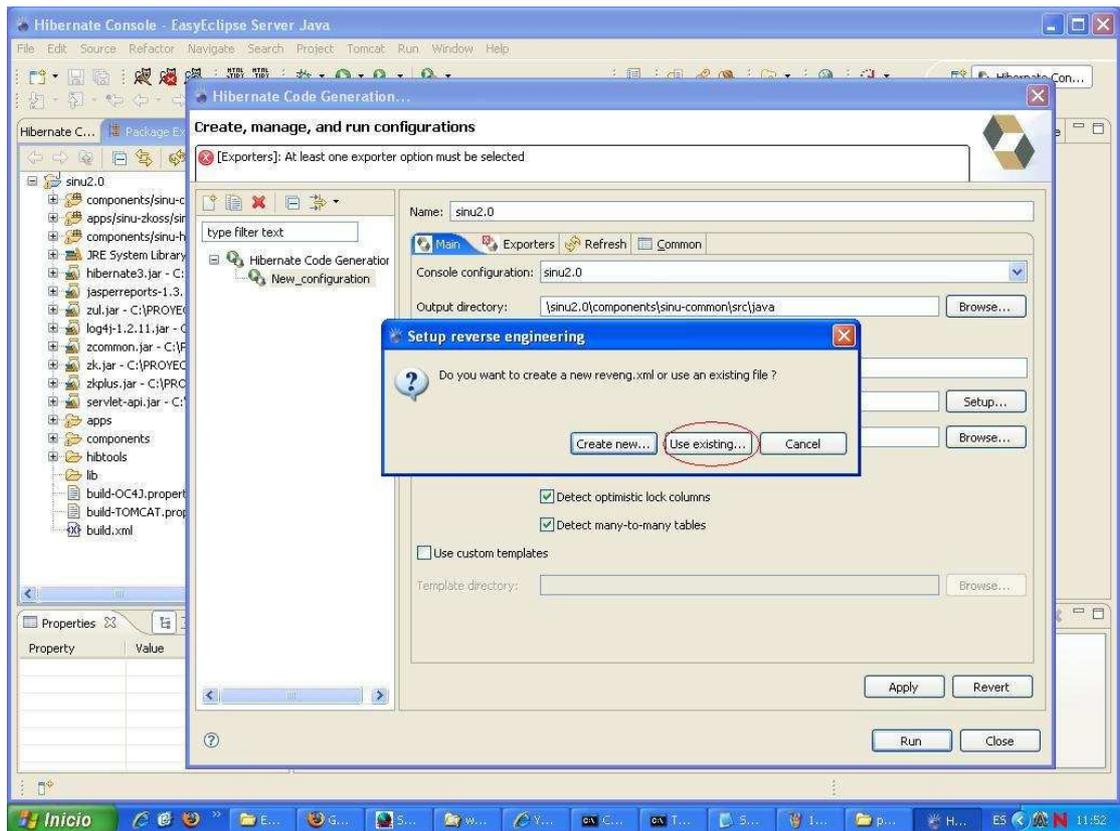


Damos clic en OK

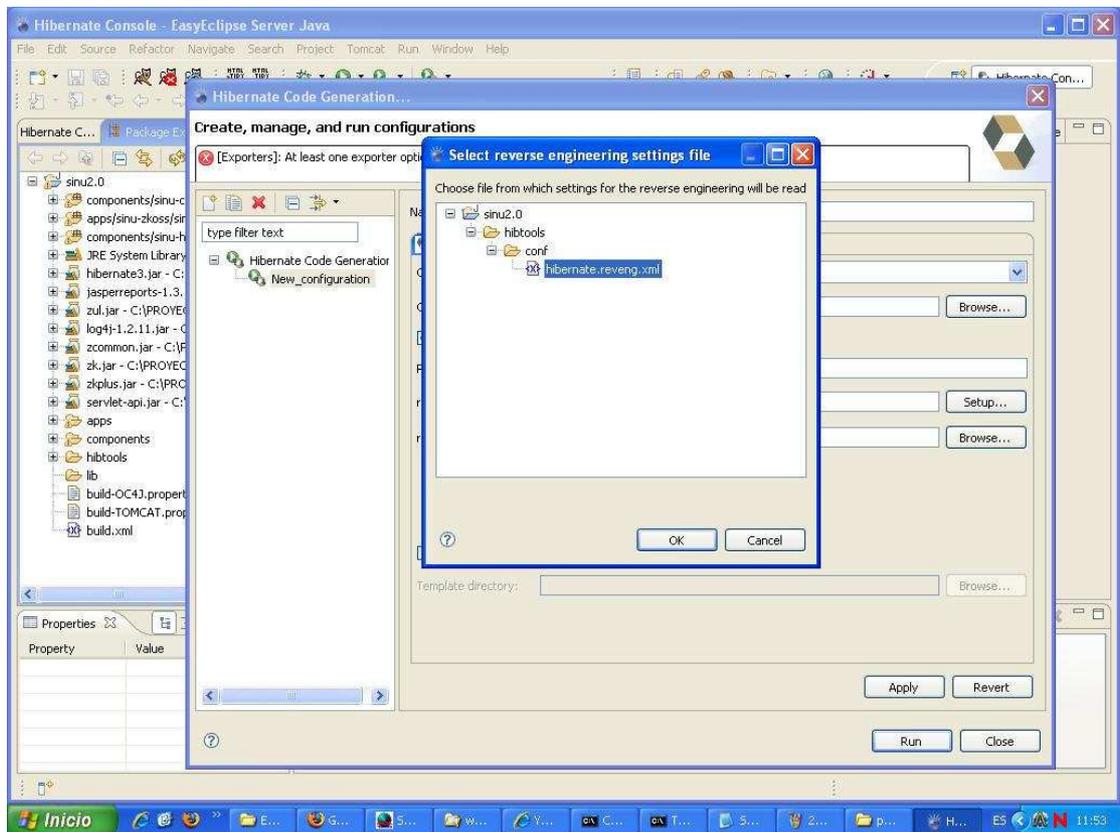
Llenamos el Package, según lo siguiente:



Damos clic en Setup, de reveng.xml y seleccionamos Use existing...

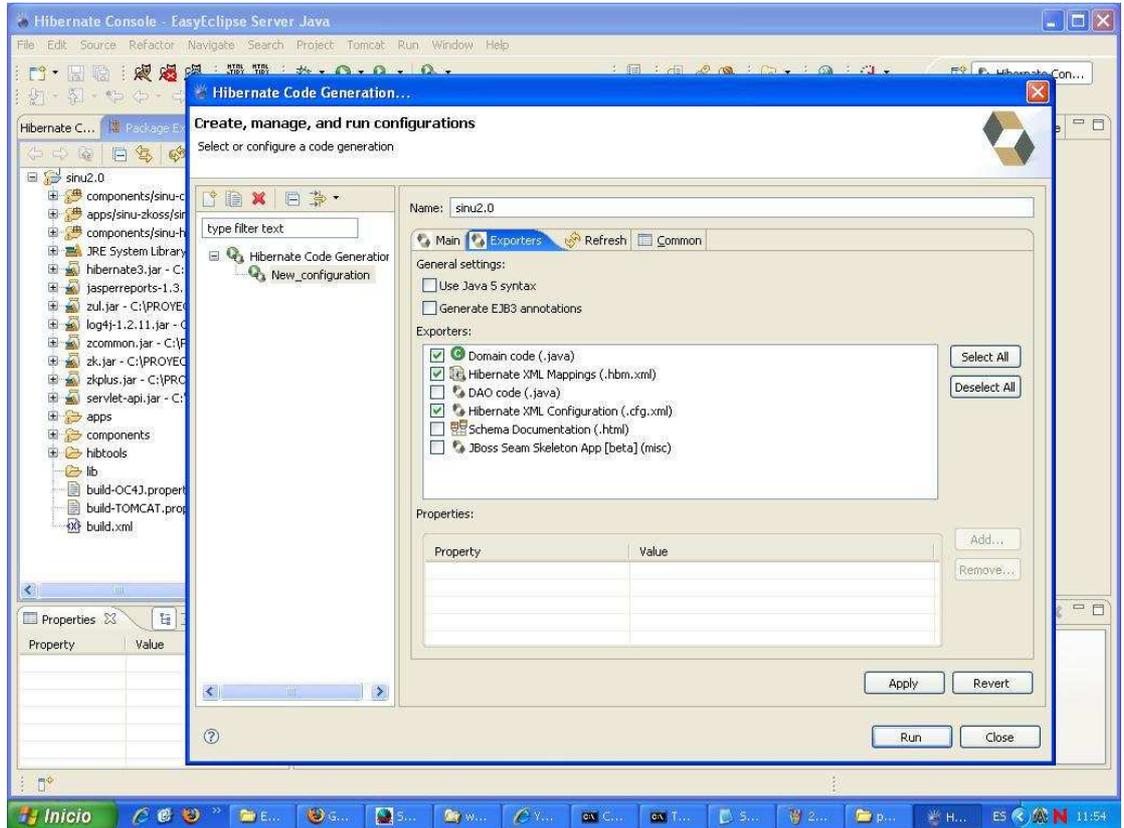


Seleccionamos hibernate.reveng.xml

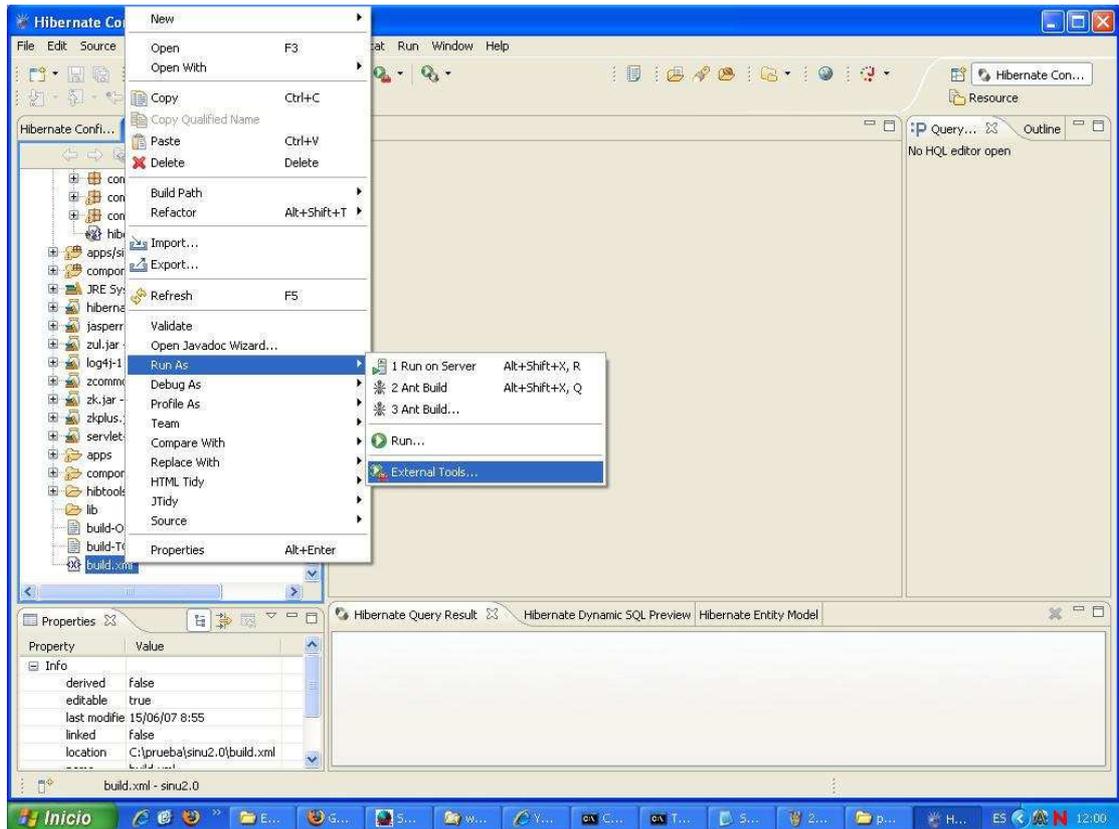


Optimimos OK

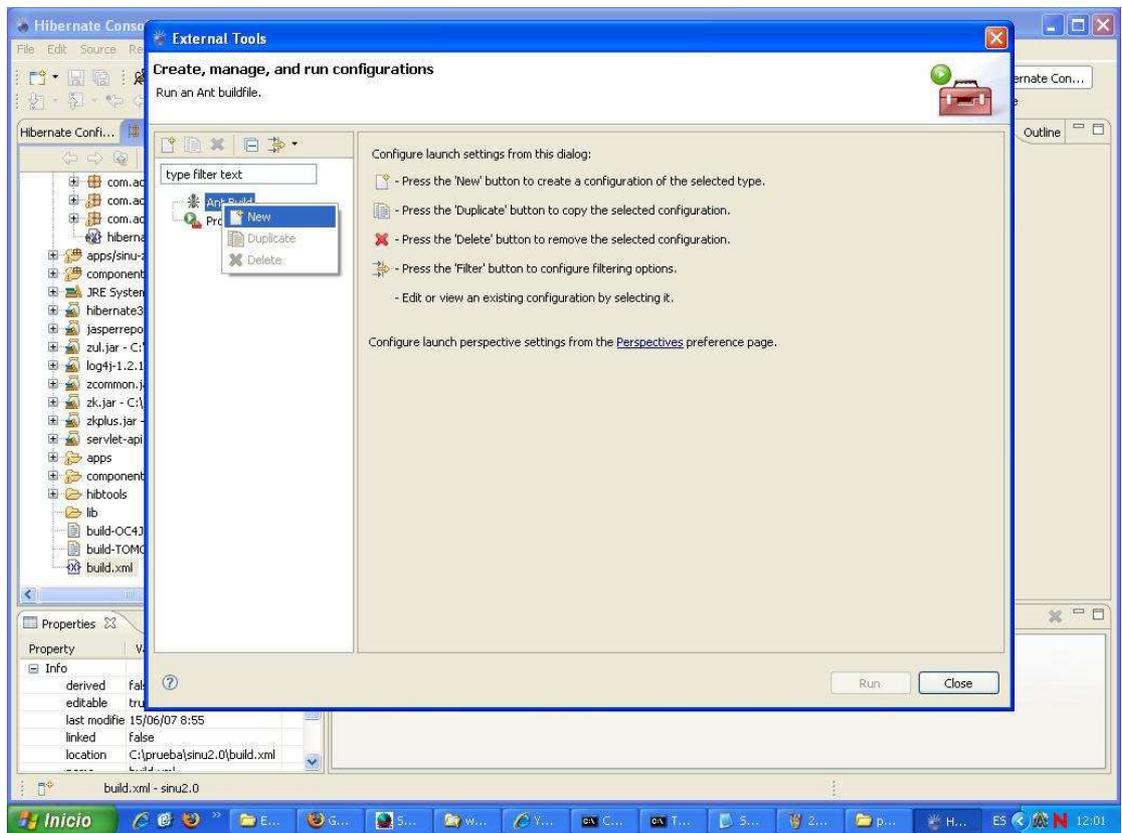
En el siguiente Tab, activamos los check box que aparecen:



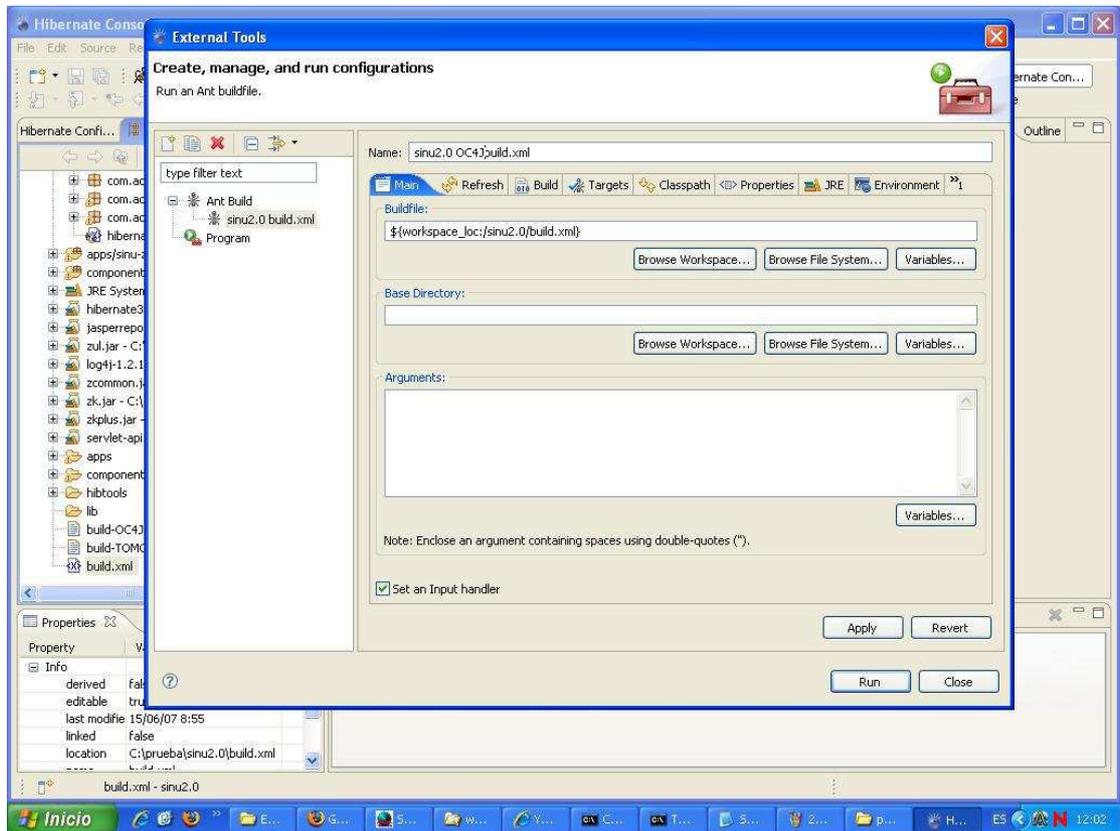
Ahora configuramos el build.xml, dar clic derecho en el mismo y seleccionar Run As... y escogemos External Tools...



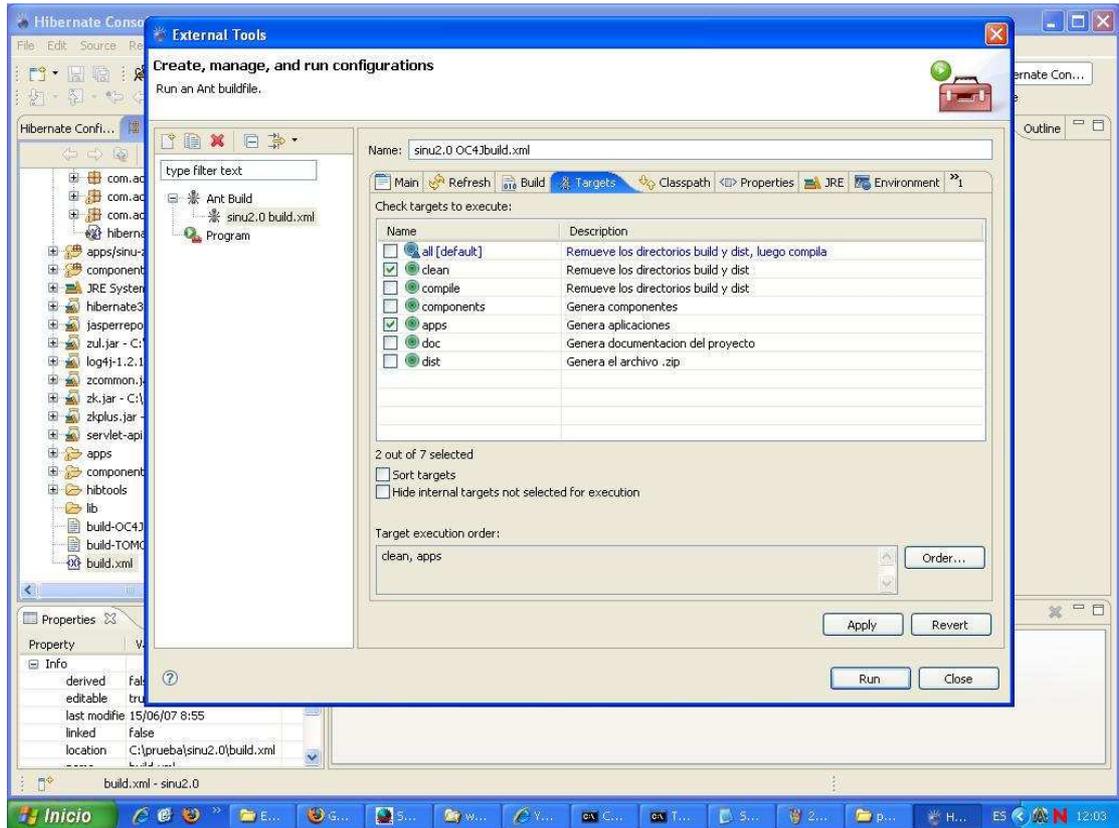
Clic derecho en Ant Build y seleccionamos New



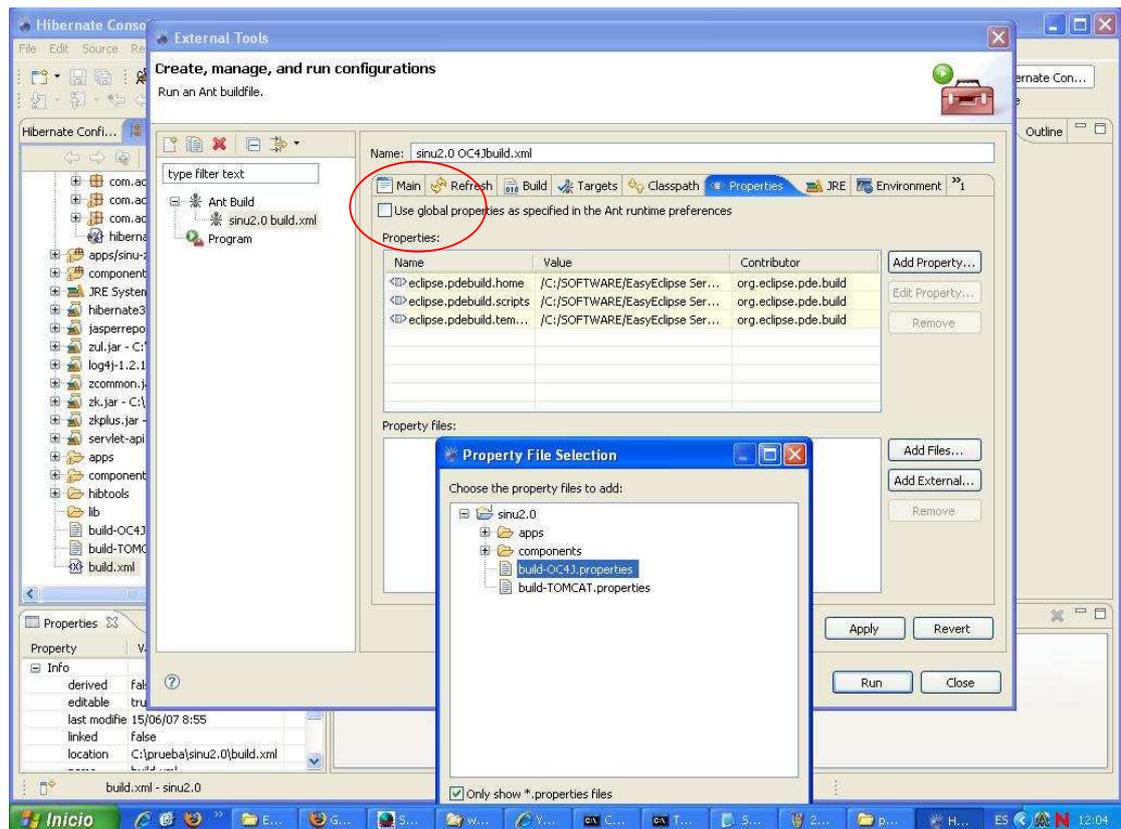
Digitamos lo siguiente en Name



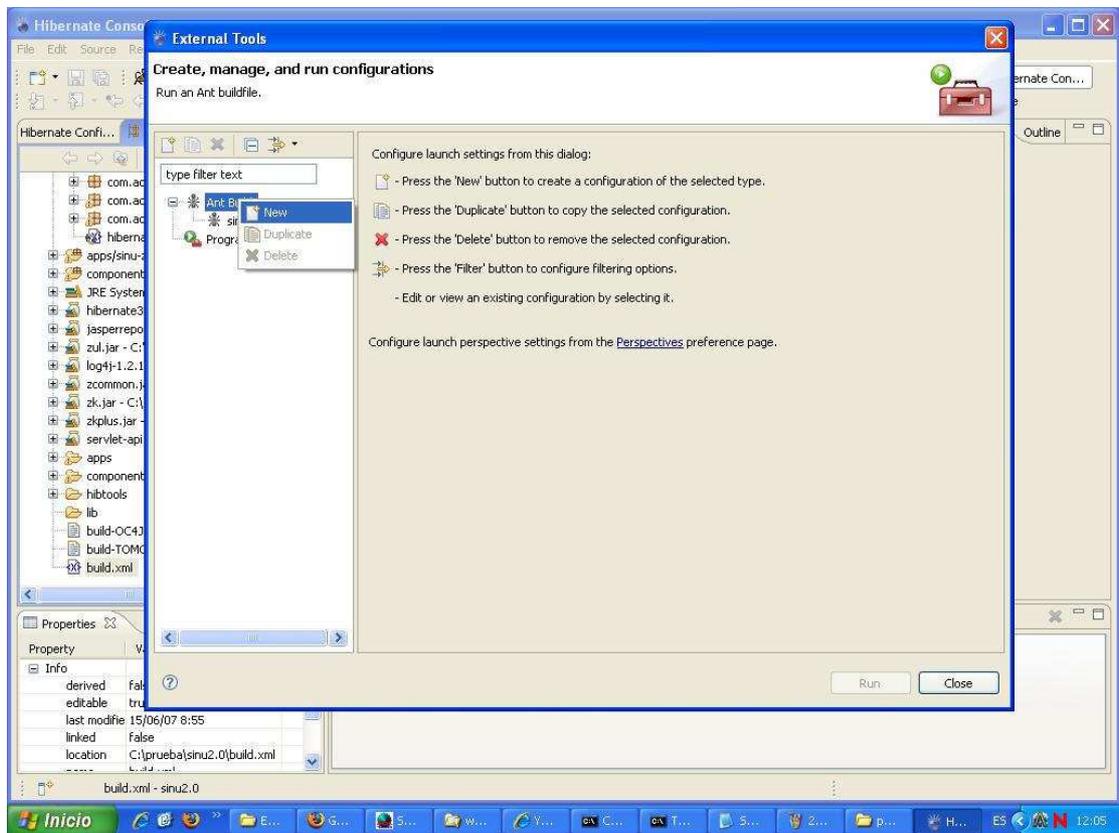
En el Tab Targets, activamos los siguientes check box



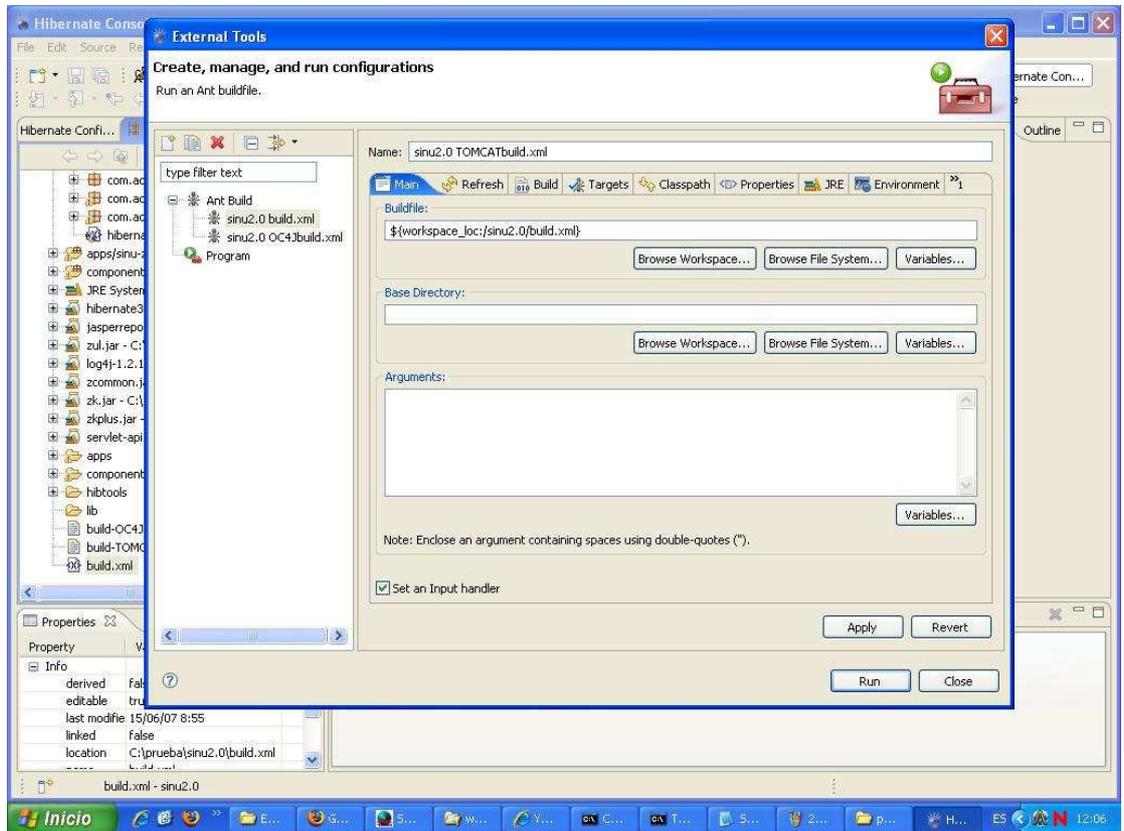
En el Tab Properties, desactivamos el check box Use global properties..., y en Add Files..., seleccionamos build-OC4J.properties



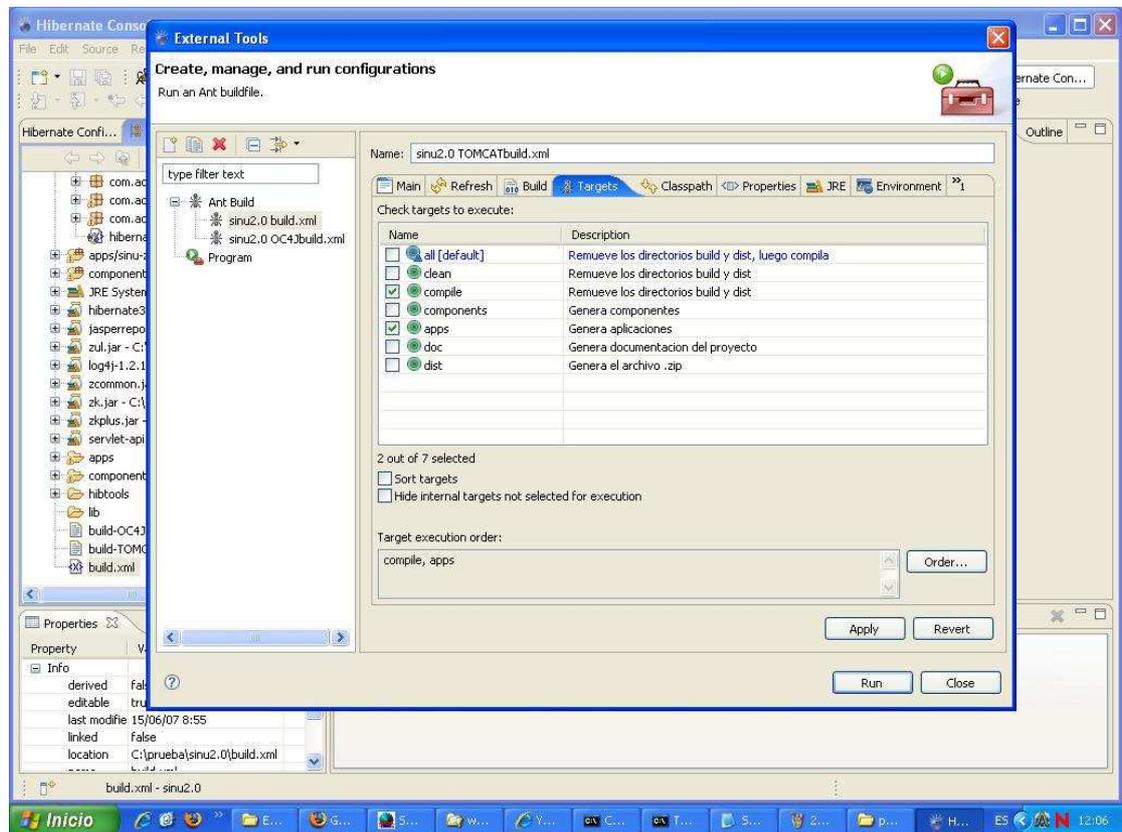
Aplicamos el mismo proceso del numeral 6.27



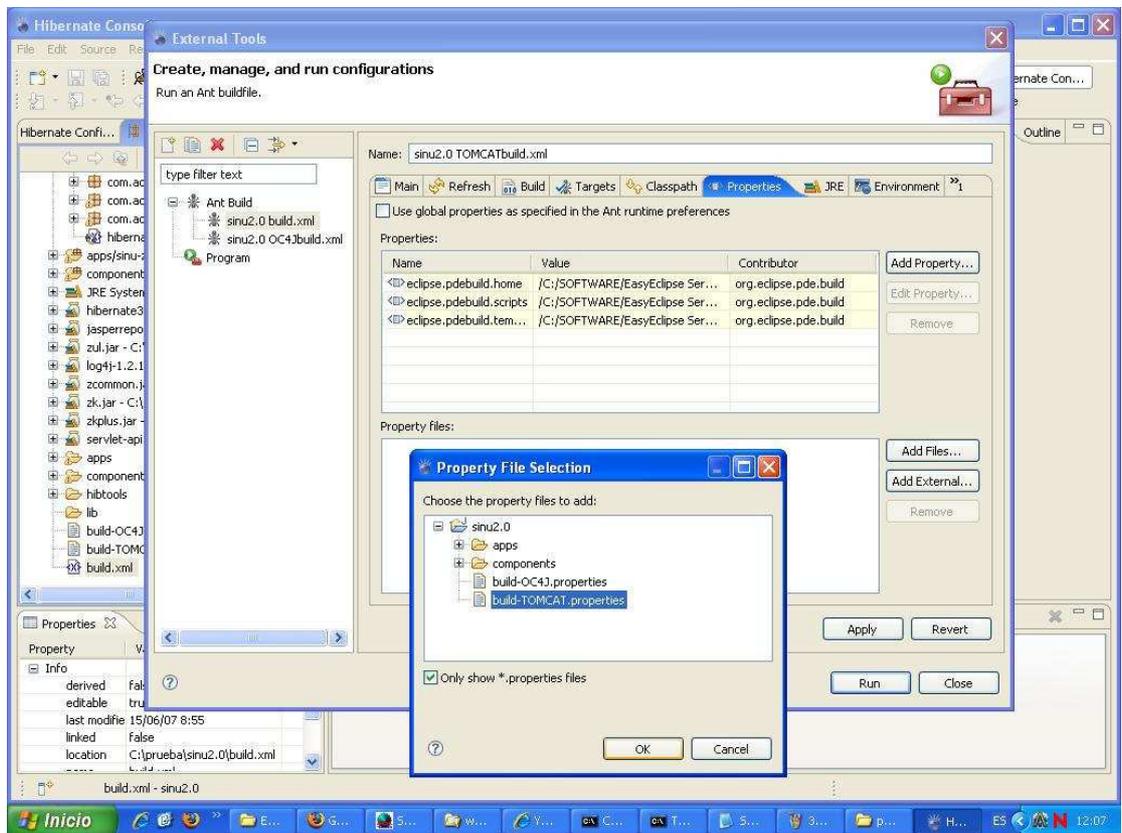
Lo denominamos TOMCAT, y digitamos la información al igual que en los numerales anteriores.



Configuramos igual que el numeral 6.28



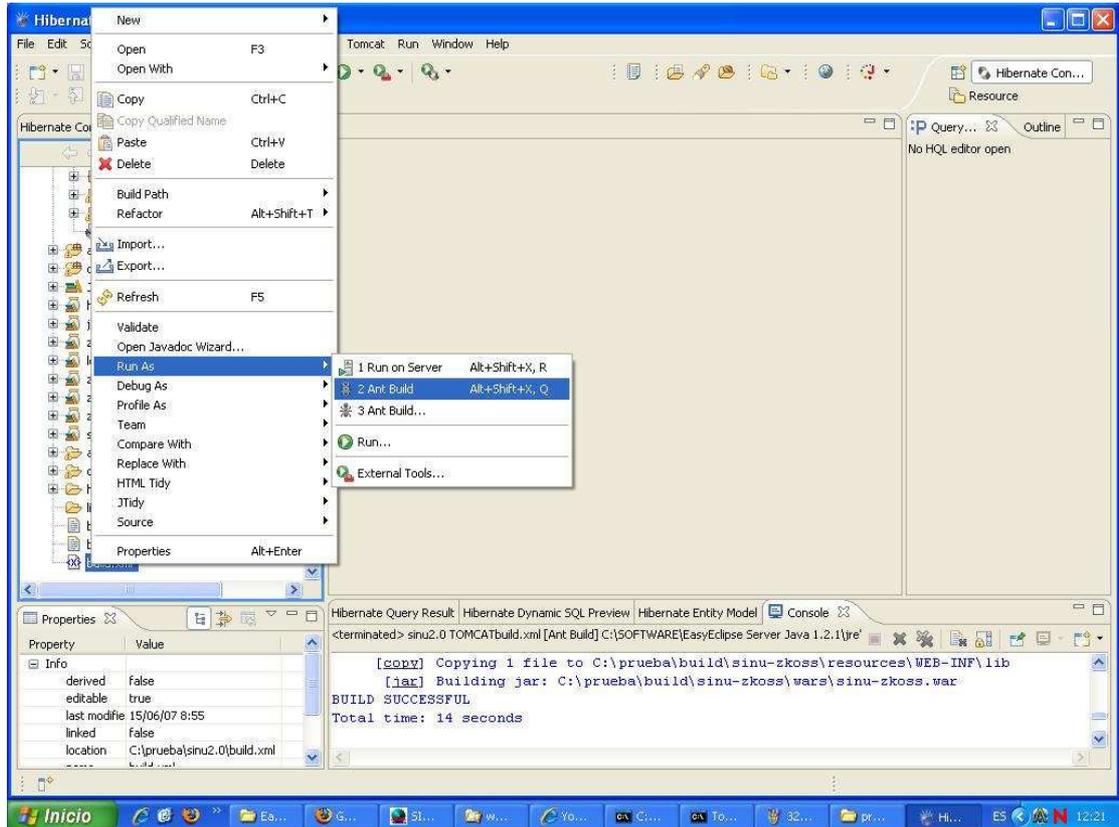
A diferencia de la configuración anterior, aquí seleccionamos build-TOMCAT.properties



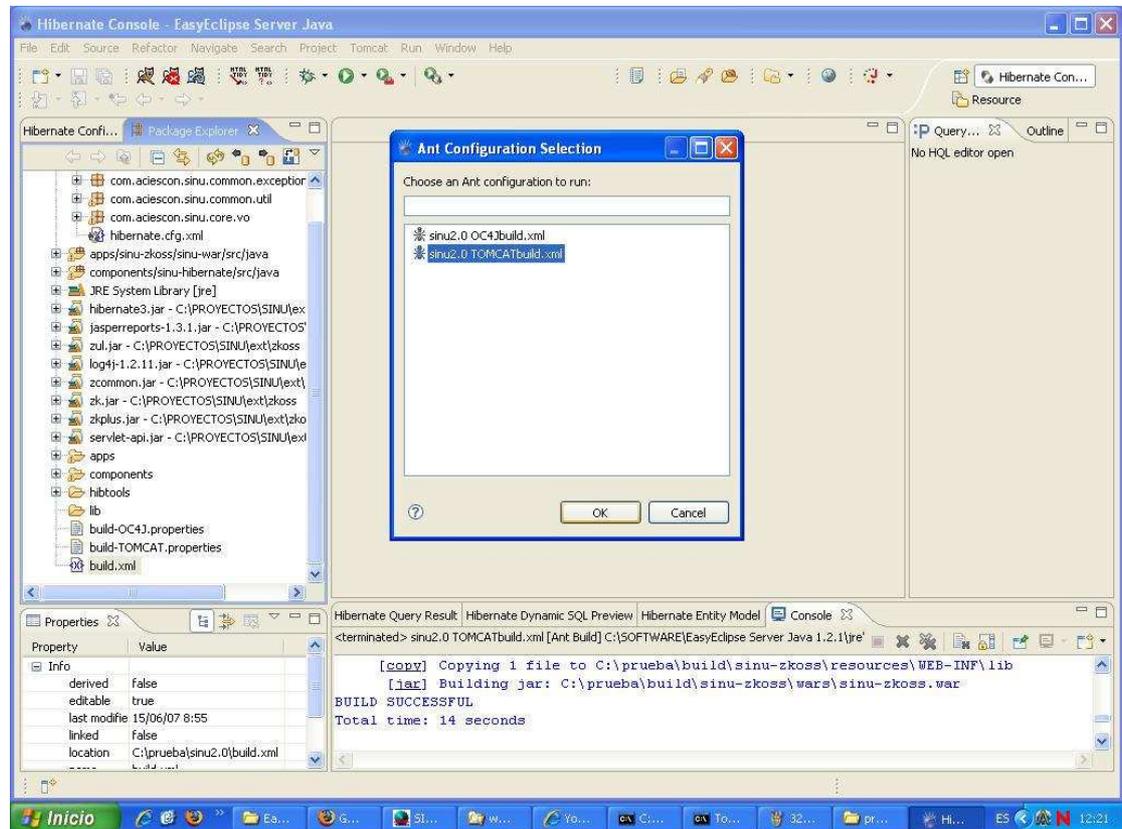
Finalmente seleccionamos Run

7. Compilar y ejecutar el proyecto

7.1 Clic derecho en build.xml, y seleccionar la opción 2 Ant Build



7.2 Seleccionamos sinu2.0 TOMCATbuild.xml



NOTA:

El proyecto genera un sinu-zkoss.war en C:\PROYECTOS\SINU\build\sinu-zkoss\wars, el cual debe ser copiado y pegado en la carpeta webapps; y luego subir el TOMCAT.

Para subir el TOMCAT, se ejecuta el archivo startup.bat en C:\SOFTWARE\Tomcat\bin y para bajarlo el shutdown.bat del mismo directorio.

Después de compilar y generar sinu-zkoss.war, siempre que se debe eliminar la carpeta sinu-zkoss en C:\SOFTWARE\Tomcat\webapps.