

-----	
---  myProyecto  -----	Ing. Ariel Alberto Bruno, 2012
-----	
---  Java 6.31 (Swing, Servlet 2.5, JSP 2.1)  -----	
-----	
---  Eclipse 3.7.2	-----
---  NetBeans 7.1.2	-----
-----	
---  Apache Tomcat 7.0.27	-----
---  HSQldb 1.8.1.3 / MySQL 5.1.13	-----
---  Hibernate 3.6.9	-----
---  JasperReports 4.5.1, iReport 4.5.1	-----
---  JavaHelp 2.0.05	-----
---  ProGuard 4.2	-----
---  Apache Ant 1.8.3	-----
-----	

## DIRECTORIOS

workspace: Espacio de Trabajo de Eclipse.

```

|
+- LEAME.txt          -> Este Archivo en texto plano.
+- LEAME.pdf          -> Este Archivo en formato PDF.
+- z Otros            -> Directorio de Archivos Extras que pueden necesitarse.
+- .metadata          -> Directorio de Archivos de Configuración de Eclipse.
+- jdk                -> Directorio del JDK Java, necesario sólo si el Cliente no lo posee.
+- jre                -> Directorio del JRE Java, necesario sólo si el Cliente no lo posee.
+- jvm.bat            -> Archivo de Configuración de JDK y JRE.
+- jar                -> Directorio con Aplicaciones necesarias para el desarrollo.
|
| +- ant              -> Directorio de Apache Ant: Herramienta de Automatizado.
| +- lib              -> Directorio de Librerías utilizadas por "src", "bin" y "hlp".
| +- opt              -> Directorio de ProGuard: Herramienta de Optimización.
|
+- myProyecto         -> Directorio del Proyecto Base generado desde Eclipse.
|
| +- prj              -> Directorio del Proyecto generado desde Eclipse.
|
|   +- .classpath     -> Configuraciones de Eclipse.
|   +- .project       -> Configuraciones de Eclipse.
|   +- src            -> Directorio de Código Fuente.
|
+- myProyectoSwing    -> Directorio de la UI del Proyecto Base generado desde Eclipse.
|
| +- LEAME.txt        -> Archivo de Información de myProyecto.
|
| +- dgr              -> Directorio de Diagramas de Base de Datos y Clases.
| +- exe              -> Directorio del Paquete con Entregable de la Aplicación.
|
| +- ant              -> Directorio con Archivo que apunta a Apache Ant.
| +- ant.bat          -> Ejecución de Ant en Consola, posibilitando elección.
| +- antProcesar.bat  -> Generación de Ejecutable a partir del Código Fuente.
| +- antRun           -> Ejecución del Ejecutable.
|
| +- bat              -> Directorio de Archivos Batch para los Procesos de myProyecto.

```

```

| +- batch.bat          -> Ejecución de Batch en Consola, posibilitando elección.
| +- batchProcesar.bat  -> Generación de Ejecutable a partir del Código Fuente.
| +- batchRun.bat       -> Ejecución del Ejecutable.
| +- batchLimpiar.bat   -> Eliminación de todo lo generado por batchProcesar.bat.
| |
| +- opt                -> Configuración para optimizar la Aplicación con ProGuard.
| |
| +- prjNetBeans        -> Directorio del Proyecto generado desde NetBeans.
| +- prj                -> Directorio del Proyecto generado desde Eclipse.
| |
| +- .classpath         -> Configuraciones de Eclipse.
| +- .project           -> Configuraciones de Eclipse.
| |
| +- src                -> Directorio de Código Fuente.
| +- bin                -> Directorio de Código Semicompilado.
| +- dat                -> Directorio de Base de Datos.
| +- rpt                -> Directorio de Reportes.
| +- hlp                -> Directorio de Ayuda.
| +- doc                -> Directorio de Documentación de JavaDoc.
| |
| +- myProyecto.jar     -> Ejecutable.
| +- myProyectoOptimizado.jar -> Ejecutable Optimizado.
| +- myProyectoFirmado.jar -> Ejecutable Firmado.
| +- lib                -> Directorio de Librerías utilizadas por Ejecutables
| |                      (copiadas desde "\jar\lib").
| +- manifiesto.mf      -> Archivo de Manifiesto a incluir en los Ejecutables.
| |
| +- .keystore          -> Almacén de Claves para Firmar la Aplicación.
| +- myProyectoCertificado.crt -> Certificado del Autor.
| |
| +- myProyectoSwing.bat -> Archivo de Ejecución Alternativa de la Aplicación.
| +- myProyectoSwing.sh  -> Archivo de Ejecución Alternativa de la Aplicación.
| +- myProyectoSwing.ini -> Archivo de Configuraciones de la Aplicación.
| +- myProyectoSwingExe.ini -> Archivo de Configuraciones de la Aplicación Final.
| +- myProyectoSwing.lic -> Licencia para poder Ejecutar la Aplicación.
| |
| +- build.xml          -> Configuración de los Procesos Automáticos para Ant.
|
+- myLicencia          -> Directorio del Generador de Licencia, generado desde Eclipse.
| (Estructura de Subdirectorios equivalente a la de myProyectoSwing).
|
+- myMutual            -> Directorio del Proyecto myMutual, generado desde Eclipse.
| (Estructura de Subdirectorios equivalente a la de myProyecto).
|
+- myMutualSwing       -> Directorio del Proyecto myMutualSwing, generado desde Eclipse.
| (Estructura de Subdirectorios equivalente a la de myProyectoSwing).
|
+- myProyectoWeb       -> Directorio del Proyecto myProyectoWeb, generado desde Eclipse.

```

---

## DESARROLLO, COMPILACIÓN Y EJECUCIÓN

---

myProyectoSwing fue realizado para ejecutarse con Java, usando como Herramientas de Desarrollo las IDE Eclipse y NetBeans (el Diseño Visual de las GUIs sólo pueden verse desde NetBeans). Para la Persistencia se usó Hibernate, junto a una Base de Datos HSQldb en la Etapa de Desarrollo y MySQL en la Etapa de Producción. Los Reportes se crearon con iReport.

La Ayuda utiliza JavaHelp. Para la Optimización del Código Fuente se usó ProGuard. Además la propia herramienta JavaDoc generó la Documentación.

El Código Fuente Java puede desarrollarse como siempre desde cualquier Editor de Textos, aunque es preferible usar una de estas IDEs que brindan muchas ayudas. Para la Edición del Código Fuente se recomienda Eclipse, y para la Edición de las GUIs se recomienda NetBeans.

Para la GUI, al haber sido creada desde NetBeans, esta IDE no sólo creó los Archivos Fuente (.java), sino que para cada uno también generó un Archivo (.form) que contiene en Formato XML la Estructura de la Ventana (Clase JFrame u otra). Si se llegase a Modificar una Clase Visual con alguna herramienta que no sea NetBeans, debería actualizarse manualmente este XML, ya que si posteriormente se editara esta misma Clase desde NetBeans, se perderían los cambios realizados (ya que NetBeans siempre verifica este XML al cargar la Clase). Por lo tanto, es altamente recomendable usar NetBeans para las modificaciones de estas Clases.

Desde cualesquiera de estas dos IDEs puede Compilarse y Ejecutarse el Proyecto. Aunque además se crearon una serie de Archivos Batch (.bat) en el Directorio "workspace\myProyectoSwing\bat", que independizan el proceso, pudiendo Compilar, Empaquetar, Optimizar, Firmar, Documentar y Ejecutar myProyectoSwing sólo ejecutando estos Archivos. O también existe la alternativa de utilizar Ant, que puede ejecutarse usando como entrada al archivo build.xml, que se lo armó para realizar todo este proceso.

-----  
DESARROLLO COMPLETO DESDE CONSOLA DE COMANDOS  
-----

En "workspace\myProyectoSwing\bat" se encuentran ordenados cada uno de los Archivos por Lotes (.bat) que permiten el desarrollo completo del Proyecto. Estos son:

01-inicializar:  
-----

Se Setea en la Variable "Path" el Directorio del Compilador de Java (javac) y de la Máquina Virtual (javaw), que se los suministra junto a la Aplicación, en el directorio "workspace". El Compilador de Java es necesario si se configuró la Aplicación para Compilar los Reportes en Tiempo de Ejecución, sino sólo es necesaria la Máquina Virtual. Se Indexa la Ayuda. Esto sirve para la Pestaña de Búsqueda. Lo que se hace es revisar los Archivos HTML de Ayuda y generar una pequeña Base de Datos que sirva para hacer las Búsquedas más rápidas y eficientes.

02-generarKeyStore:  
-----

Se genera el Almacén de Claves, necesario para Firmar la Aplicación.  
Este paso no es necesario ejecutarlo más de una vez a lo largo del desarrollo del Proyecto.

03-generarCertificado:  
-----

Se genera el Certificado del Autor.  
Este paso no es necesario ejecutarlo más de una vez a lo largo del desarrollo del Proyecto.  
Igualmente, el Certificado en formato ".crt" no es necesario.

04-compile:  
-----

Se Compila la Aplicación, generando los Archivos Semicompilados (Bytecodes, .class) a partir del Código Fuente (.java).  
Además se Copian los Archivos de Imágenes, Configuración y Mapeo desde el Directorio Fuente ("workspace\myProyectoSwing\prj\src") al Directorio de Compilación ("workspace\myProyectoSwing\prj\bin").

## 05-empaquetar:

-----

Se Empaqueta la Aplicación, generando el Archivo Ejecutable (.jar) a partir de los Archivos Semicompilados (Bytecodes, .class).  
Además, se Copian las Librerías que necesita la Aplicación. Para ello, se extraen del Directorio "workspace\jar\lib" (donde se encuentran las Librerías que se usan y algunas que podrían ser útiles en el futuro, ordenadas en una Estructura de Directorios) las Librerías necesarias y se las Copia en "workspace\myProyectoSwing\prj\lib" (donde están sólo las Librerías necesarias juntas, sin crear ningún Subdirectorio).

## 06-optimizar:

-----

Se Optimiza la Aplicación utilizando la herramienta Proguard, que se encuentra en "workspace\jar\opt". Los pasos que se realizan son:

- 1- Encogimiento: Se analizan Bytecodes y se quitan Clases, Propiedades y Métodos No Usados.
- 2- Optimización: Se evalúan Expresiones Constantes; se quitan Accesos a Atributos, Llamadas a Métodos, Ramificaciones y Comparaciones Innecesarios, Bloques de Código No Usados, Campos de Sólo Escritura y Parámetros de Métodos No Usados, Campos, Parámetros y Valores Devueltos Constantes; se combinan Bloques de Código Idénticos; se reducen las Asignaciones de Variables; se hacen Métodos Privados, Estáticos y Finales cuando sea posible; se hacen Clases Estáticas y Finales cuando sea posible, se sustituyen los Interfaces que tienen una sola Implementación.
- 3- Ofuscamiento: Se quita cualquier Información de Debug y se sustituyen todos los Nombres posibles por Secuencias de Caracteres sin significado, haciendo mucho más difícil la Ingeniería Inversa del Código. Esto además reduce aún más el Código. Aunque muchas Clases No pueden Ofuscarse, como por ejemplo las Clases Persistentes, que necesitan consistencia con los Archivos de Mapeo de Hibernate.
- 4- Preverificación: Se asegura que el Código resultante no escape de los Límites de Ejecución de la Máquina Virtual.

## 07-firmar:

-----

Se Firma la Aplicación, generando el .jar Firmado a partir del .jar Optimizado.

## 08-generar:

-----

Se genera el Directorio "workspace\myProyectoSwing\exe", y dentro de éste todos los Archivos necesarios para la Ejecución en Producción de la Aplicación (Ejecutable, Librerías, Base de Datos, Reportes, Ayuda). Es decir, este Directorio es la Aplicación Entregable Final.

## 09-documentar:

-----

Se genera el Directorio "workspace\myProyectoSwing\prj\doc", y dentro de éste todos los Archivos de Documentación de la Aplicación, generado por JavaDoc.

## 10-ejecutar:

-----

Se Ejecuta la Aplicación, usando para ello los Archivos Semicompilados (Bytecodes, .class) generados en el paso de Compilación y las Librerías del Directorio "workspace\jar\lib".

## 11-ejecutarJar:

-----

Se Ejecuta la Aplicación, usando para ello el Archivo Ejecutable (.jar) y las Librerías del Directorio "workspace\myProyectoSwing\prj\lib", generados en el paso de Empaquetado.

## 12-ejecutarJarOptimizado:

Se Ejecuta la Aplicación, usando para ello el Archivo Ejecutable (.jar) generado en el paso de Optimización, y las Librerías del Directorio "workspace\myProyectoSwing\prj\lib", generadas en el paso de Empaquetado.

## 13-ejecutarJarFirmado:

Se Ejecuta la Aplicación, usando para ello el Archivo Ejecutable (.jar) generado en el paso de Firmado, y las Librerías del Directorio "workspace\myProyectoSwing\prj\lib", generadas en el paso de Empaquetado.

## 14-ejecutarExe:

Se Ejecuta la Aplicación Final, usando todo lo que necesita del Directorio "workspace\myProyectoSwing\exe".

## 15-procesar:

Se realiza todo el Proceso completo: a partir del Código Fuente, se realiza la Inicialización, Compilación, Empaquetado, Optimización, Firmado, Documentación y Generación de la Aplicación Final.

## RESUMEN DE PASOS

Por lo tanto, una vez que se tenga el Código Fuente, sólo deberá ejecutarse el Archivo "batchProcesar.bat", que se encuentra en el Directorio Base ("workspace\myProyectoSwing"). Éste llamará a "15-procesar.bat", creándose automáticamente el Directorio "workspace\myProyectoSwing\exe", donde se encontrarán todos los Archivos necesarios para la Ejecución en Producción, que son los siguientes:

myProyectoSwing.jar: Archivo Ejecutable,  
myProyectoSwing.bat: Archivo por Lotes que Ejecuta "myProyectoSwing.jar" (opcional),  
myProyectoSwing.sh : Archivo Shell Script que Ejecuta "myProyectoSwing.jar" (opcional),  
myProyectoSwing.ini: Archivo de Configuraciones,  
myProyectoSwing.lic: Archivo de Licencia necesario para la Ejecución de la Aplicación,  
lib: Directorio donde se encuentran todas las Librerías necesarias,  
dat: Directorio que contiene la Base de Datos en HSQLDB (recomendada usar MySQL),  
rpt: Directorio donde se encuentran los Archivos de Reportes en Formato XML,  
hlp: Directorio que guarda la Ayuda de la Aplicación,  
log: Directorio con los Archivos de Log, uno por día (creación en Tiempo de Ejecución).

Otra opción para realizar este proceso completo es ejecutar "antProcesar.bat", que también se encuentra en el Directorio Base ("workspace\myProyectoSwing"). Éste llamará a la Aplicación "Ant" ("workspace\myProyectoSwing\prj\ant") con el parámetro "generar", correspondiendo este Objetivo de Ant a lo que necesitamos realizar.

## NOTAS ADICIONALES

"myProyectoSwing.bat" (Windows) o "myProyectoSwing.sh" (Linux) ejecutará "myProyectoSwing.jar", usando la Máquina Virtual que trae la Aplicación si se la Copió, o sino usando la Máquina Virtual que está Instalada en la Computadora. Este Archivo es

opcional, pero la Ejecución de la Aplicación desde éste es imprescindible si no se tiene instalado el jre o el jdk en la Computadora (de ser así, deberá haberse copiado la Máquina Virtual que trae la Aplicación).

Si no se incluye la Base de Datos (directorio "dat" si se está utilizando HSQLDB o si no se creó la Base de Datos si se está usando MySQL), Hibernate crea automáticamente una Base de Datos vacía con la Estructura que le provee la Aplicación (aunque no se podrá Loguear en el Sistema al no poseer ningún Usuario).

Si no se incluye el directorio "rpt" (o si no se posee el JDK de Java si compilamos los Reportes en Tiempo de Ejecución o sino si no poseemos el JRE), no se podrán Imprimir los Reportes, aunque se podrá seguir usando la Aplicación.

Si no se incluye el directorio "hlp", no se podrá Visualizar la Ayuda, aunque se podrá seguir usando la Aplicación.

En el Subdirectorio "log" del Directorio de Ejecución se encontrarán los Archivos de Logs (llamados "log\_yyyyMMdd.log", es decir, uno por día), donde se registrarán todos los Sucesos Importantes que se produzcan con la Utilización de myProyectoSwing. El registro de Logs podrá ser desactivado seteando el parámetro correspondiente desde el Archivo de Configuraciones "myProyectoSwing.ini".

El directorio "doc" no se incluye en la Aplicación Final, ya que es documentación útil sólo a los desarrolladores.

Para myLicencia y myMutual el proceso de desarrollo es el mismo.

-----