

T7.Planning.

Índice

1. Introducción	3
2. Objetivos de aprendizaje	3
3. Agenda	3
4. Lista de requerimientos (Product Backlog)	3
4.1. Lectura: Teoría tema7	3
4.2. Generación de paquete de instalación desde el IDE: .jar	3
4.3. Creación de instaladores con NSIS	4
4.3.1. Creación de script para instaladores Windows I: Basic	4
4.3.2. Creación de script para instaladores Windows II: HeaderBitmap	4
4.3.3. Creación de script para instaladores Windows III: MultiLanguage	4
4.3.4. Creación de script para instaladores Windows IV: StartMenu	5
4.3.5. Creación de script para instaladores Windows V: WelcomeFinish	5
4.3.6. Creación de script para instaladores Windows VI: LanzaAyuda	5
4.4. Firma digital de ficheros JAR e instalador: AppHotel.....	5
4.4.1. Artículo acerca del algoritmo de cifrado MD5	5
4.4.2. Distribución aplicación AppHotel (instalador)	6
4.4.3. Distribución aplicación AppAgenda vía Web (Jpro).....	6
5. Criterio de evaluación	6

1. Introducción

Aprenderemos los distintos procedimientos de distribución de una aplicación. Aprenderemos a empaquetar aplicaciones para su distribución preparando paquetes auto instalables con asistentes incorporados. Aprenderemos a Desplegar y distribuir aplicaciones en distintos ámbitos de implantación verificando su comportamiento y realizando las modificaciones necesarias.

Utilizaremos el framework Eduscrum. Los alumnos trabajaréis en equipos y preparareis una presentación para mostrar lo que habéis logrado.

2. Objetivos de aprendizaje

- ✓ El proceso de instalación. Tipos, completa, típica y personalizada. Actualizaciones.
- ✓ Componentes de una aplicación. Empaquetado.
- ✓ Instaladores. Tipos y características.
- ✓ Paquetes autoinstalables.
- ✓ Herramientas para crear paquetes de instalación.
- ✓ Personalización de la instalación, logotipos, fondos, diálogos, botones, idioma, entre otros.
- ✓ Asistentes de instalación y desinstalación
- ✓ Interacción con el usuario.
- ✓ Ficheros firmados digitalmente.
- ✓ Instalación de aplicaciones desde un servidor web.
- ✓ Distribución de aplicaciones vía servidores web

3. Agenda

- ✓ Miércoles 03/03/2021
Reunión de planificación (1 día / 2 horas).
- ✓ Jueves 04/03/2021 a martes 16/03/2021
Sprint (7 días / 16 horas).
- ✓ Martes 16/03/2021
Entrega.

4. Lista de requerimientos (Product Backlog)

4.1. Lectura: Teoría tema7

Cada alumno debe leer la teoría del tema7. Se deberá referenciar adecuadamente los enlaces en los marcadores.

Definición de Hecho: Has leído, entendido el texto y respondido a todos los *RESPONDE* del documento.

4.2. Generación de paquete de instalación desde el IDE: .jar

Cada alumno, siguiendo las indicaciones del punto 6.1 del tema, revisará la aplicación LanzaAyuda que ya tenemos desarrollada del tema anterior y hemos especificado la clase main así como construido el jar. Se analizará el contenido del directorio **dist**. Se verificará que se han incluido todos los recursos de la aplicación, todas las librerías y que el archivo **manifest** contiene la entradas: Main-Class: paquete.MainClass y Class-Path: lib/javaahelp-2.0.05.jar, etc...

Nota: Antes de distribuirlo se deberá probar el correcto y completo funcionamiento de la aplicación LanzaAyuda fuera del IDE Netbeans. Para ello desde la consola ejecuta:

```
#java -jar C:\Users\usuario\ruta-jar\LanzaAyuda.jar
```

Es posible que las referencias a los archivos en ejecución no se devuelvan igual trabajando desde el entorno de desarrollo que fuera del mismo. Por lo que si no funciona correctamente se deberá modificar en el código las referencias como en el siguiente ejemplo, que las obtiene desde el jar:

```
URL url = new URL("jar:file:LanzaAyuda.jar!/lanzaayuda/help/jhelpset.hs");
```

Definición de Hecho: Cada alumno ha ejecutado LanzaAyuda.jar desde fuera del entorno de desarrollo y le ha funcionado.

4.3. Creación de instaladores con NSIS

Cada miembro del equipo, siguiendo las indicaciones del punto 9 del tema, instalará la herramienta NSIS en su sistema y realizará las actividades indicadas a continuación. Para llevar a cabo la creación de instaladores seguiremos la documentación de NSIS UI Moderno en el enlace: <https://nsis.sourceforge.io/Docs/Modern%20UI%202/Readme.html>. Y se revisarán los scripts de ejemplo del directorio de instalación de la herramienta ..\NSIS\Examples\Modern UI.

4.3.1. Creación de script para instaladores Windows I: **Basic**

Cada alumno creará un directorio de su home, por ejemplo: C:\Users\Usuario\MisScript, y replicará el primer script "Basic.nsi" de los ejemplos, llama al archivo MiScript1.nsi. Analiza el código antes de compilarlo para crear el primer instalador, llámalo Instalador1.exe. Adáptalo a tu gusto y prueba el instalador generado en tu sistema Windows.

Definición de Hecho: Cada alumno habrá generado, compilado y probado el script. La prueba supone lanzar la instalación en el SO y verificar que ha realizado lo que pretendía.

4.3.2. Creación de script para instaladores Windows II: **HeaderBitmap**

Cada alumno replicará el script "HeaderBitmap.nsi" de este tutorial. Llamará al archivo MiScript2.nsi. Analiza el código antes de compilarlo para crear el instalador, llámalo Instalador2.exe. Se adaptará entre otras cosas con la imagen de la empresa y se probará en el sistema Windows verificando que se visualiza la imagen.

Definición de Hecho: Cada alumno habrá generado, compilado y probado el script. La prueba supone lanzar la instalación en el SO y verificar que ha realizado lo que pretendía.

4.3.3. Creación de script para instaladores Windows III: **MultiLanguage**

Cada alumno replicará el script "MultiLanguage.nsi", incluyendo que el instalador copie el archivo "test.txt (se deberá crear con contenido)" en el directorio de instalación. Al ejecutar el instalador, se debe instalar el archivo test.txt en el directorio indicado. Llama al script MiScript3.nsi. Se analizará el código antes de compilarlo para crear el instalador, se llamará Instalador3.exe. Se probará el instalador generado en el sistema Windows verificando que se ha copiado el archivo correspondiente en el directorio indicado.

Definición de Hecho: Cada alumno habrá generado, compilado y probado el script. La prueba supone lanzar la instalación en el SO y verificar que ha realizado lo que pretendía.

4.3.4. Creación de script para instaladores Windows IV: StartMenu

Cada alumno replicará el script "StartMenu.nsi". Llamará al archivo MiScript4.nsi. Analizará el código antes de compilarlo para crear el instalador, lo llamará Instalador4.exe. ¿Qué funcionalidad incluye? Se adaptará a gusto y se probará en el sistema Windows verificando el funcionamiento correcto.

Definición de Hecho: Cada alumno habrá generado, compilado y probado el script. La prueba supone lanzar la instalación en el SO y verificar que ha realizado lo que pretendía.

4.3.5. Creación de script para instaladores Windows V: WelcomeFinish

Cada alumno replicará el script "WelcomeFinish.nsi". Analizará el código y lo adaptará antes de compilarlo para crear el instalador, lo llamará Instalador5.exe. Lo probará en el sistema Windows verificando el funcionamiento correcto.

Definición de Hecho: Cada alumno habrá generado, compilado y probado el script. La prueba supone lanzar la instalación en el SO y verificar que ha realizado lo que pretendía.

4.3.6. Creación de script para instaladores Windows VI: LanzaAyuda

Cada alumno creará un instalador llamado InstallAyuda.exe que instale la aplicación LanzaAyuda en Windows. Requerimientos: el script se llamará "LanzaAyuda.nsi". El instalador deberá presentar **pantalla de bienvenida**, de **selección de componentes**, de **selección de directorio de instalación** y debe crear un elemento en el **menú de inicio** llamado LanzaAyuda que apunte al archivo LanzaAyuda.jar que previamente se ha **descomprimido** (el zip de la aplicación). También deberá incluir un **desinstalador**, incluyendo un enlace en el menú de inicio, llamado UninstallLanzaAyuda.exe de tal manera que elimine los archivos y los elementos de menú. Es de gran ayuda el ejemplo2.nsi proporcionado por la herramienta.

Nota: Se requerirá usar el plugin ZipDLL para descomprimir el zip de la aplicación en el destino: https://nsis.sourceforge.io/Nsis7z_plugin. En el enlace se indica el procedimiento de uso.

Definición de Hecho: Cada alumno habrá generado, compilado y probado el script. La prueba supone lanzar la instalación en el SO y verificar que ha realizado lo que pretendía.

4.4. Firma digital de ficheros JAR e instalador: AppHotel

4.4.1. Artículo acerca del algoritmo de cifrado MD5

Cada miembro del equipo leerá el artículo <https://www.redeszone.net/2017/01/20/oracle-bloquear-todos-los-jar-firmados-md5-partir-abril/> y comentará lo más relevante de la noticia.

Definición de Hecho: Cada alumno habrá documentado los comentarios acerca de la noticia de Oracle.

4.4.2. Distribución aplicación AppHotel (instalador)

Cada alumno desde NetBeans revisará la aplicación AppHotel de su equipo en última versión y una vez construido el jar se accederá al directorio dist y analizará su contenido:

- ✓ Verificando que se han incluido todos los recursos de la aplicación, todas las librerías y que el archivo manifest contiene la entradas: Main-Class: paquete.MainClass y Class-Path: lib/Librerias.jar, etc.
- ✓ Firmará el jar de la aplicación sin certificación (autofirma). Revisa para ello el Anexo I del tema.
- ✓ Ejecutando y comprobando que la aplicación se ejecuta fuera de NetBeans.
- ✓ Creando un instalable con NSIS para instalar y ejecutar la aplicación en Windows.
- ✓ Comprobando la instalación y correcta ejecución de la aplicación en Windows.

Definición de Hecho: Cada alumno habrá realizado el empaquetado, firmado y creado el instalador de la aplicación AppHotel y habrá probado su funcionamiento.

4.4.3. Distribución aplicación AppAgenda vía Web (Jpro)

Cada alumno creará un proyecto **Gradle** para la aplicación AppAgenda, lo llamará **AppAgendaJPRO** generando el script de construcción (build.gradle) con todos los plugins y las dependencias necesarias e incluirá los archivos jpro requeridos para lanzar la aplicación vía Web.

Se revisará para el nuevo proyecto AppAgendaJPRO los directorios y ficheros generados tras la construcción del proyecto para contrastarlos con los generados al construir la aplicación con el IDE NetBeans tipo Ant.

Definición de Hecho: Cada alumno ha generado el script build.gradle, y realizado el resto de elementos necesarios para ejecutar la aplicación vía Web usando JPRO y habrá probado su funcionamiento.

5. Criterio de evaluación

Porcentaje de requerimientos pendientes (60%)

Tareas completadas.

Calidad (40%)

Características de originalidad, diseño y fiabilidad.