


Curso :	2º CFS DAM	Fecha límite :	19/11/20 14:30 (18 horas)
Trimestre :	1	Modalidad :	Individual
 <div> Práctica de Desarrollo de Interfaces 1T_3 INFORMES CON JASPERREPORTS </div>			

OBJETIVOS

El resultado primordial que persigue esta práctica es completar la visión que en materia de interfaces gráficos implementados en Java Swing y JavaFX tenga el alumno incidiendo en los interfaces de salida cuyo ejemplo más representativo es el informe. Para estos también han de observarse los fundamentos de presentación de los IU buscando el aspecto más profesional de las aplicaciones. Además se persigue que el alumno, aprenda por sí mismo a partir de la experiencia del trabajo a realizar a la vez que averigua información adicional para completar los diferentes apartados. Se propician también interacciones de aprendizaje con el resto de compañeros al tener que construir conocimiento así como una disciplina de trabajo continuado para cumplir con el plazo de entrega fijado.

1. Se han identificado los modos de generar informes: desde el código del programa o desde asistente gráfico.
2. Se han identificado distintos motores de generación de informes.
3. Se han identificado los tipos de informes y su contexto de uso: incrustados y no incrustados
4. Se ha analizado la estructura (secciones) de un informe JasperReports: desde el básico hasta los predefinidos.
5. Se ha identificado el proceso de generación de informes en JasperReports con las fases y tipos de ficheros.
6. Se han generado informes básicos a partir de una fuente de datos no persistente mediante asistentes.
7. Se han establecido filtros sobre los valores a presentar en los informes.
8. Se han incluido valores calculados, recuentos, totales y gráficos generados a partir de los datos.
9. Se han utilizado herramientas para generar el código correspondiente a los informes de una aplicación.
10. Se ha modificado el código correspondiente a los informes.
11. Se ha desarrollado una aplicación que incluye informes incrustados.

RESULTADOS Y COMPETENCIAS PERSEGUIDOS

En consonancia a los resultados de aprendizaje y competencias que el alumno debe completar cursando este módulo, este trabajo contribuye a conseguir los siguientes:

R2) Generación de interfaces a partir de documentos XML

R5) Confección de informes.

- C2) El trabajo regular y la constancia.
- C3) El aprendizaje con los demás.
- C4) El aprendizaje por sí mismo.
- C5) La autonomía responsable.
- C6) El trabajo en equipo.
- C7) El dominio de las herramientas de comunicación más avanzadas.
- C8) La selección y priorización de la información manejada.
- C9) La presentación profesional de la información requerida.

ATERRIZAJE EN LA PRÁCTICA

Hasta ahora sólo se ha visto una parte de los interfaces que forman parte de las aplicaciones: los que permiten la entrada y salida de datos por la misma pantalla. Pero, ¿qué pasa cuando se

pretende que una aplicación tenga la funcionalidad de generar informes que pueden mostrarse en formatos de salida como PDF o similares?

Tenemos por una parte la posibilidad de realizar esta tarea a mano, es decir, implementando línea a línea de código Java/XML o utilizar entornos que incluyan algún tipo de asistente visual que permita construir fácilmente la estructura para posteriormente ir perfilando el informe de manera que sea reconocible por una aplicación Java que tire del informe cuando lo requiera.

En el caso concreto de esta práctica, se busca en primer lugar aprovechar la aplicación anterior **Visor** (versión Swing y JavaFX) de forma que genere unos informes que presenten datos y gráficos.

NO DEBES CONTINUAR SIN TENER ESTO CLARO

► **JasperReports:** es una librería (no una aplicación) diseñada para la generación de informes, implementada en su momento en lenguaje Java bajo licencia GNU LGPL. Por eso es fácil que ésta pueda ser incrustada en cualquier aplicación Java que requiera informes.

► **Problemas con IntelliJ IDEA:** Jasper funciona, a día de hoy, correctamente con Eclipse (vía plugin) y NetBeans. IntelliJ IDEA no termina de ver la implementación de un plugin para esta herramienta.

► **Apache Ant:** es una herramienta (librería) usada en programación para la realización de tareas mecánicas y repetitivas, normalmente durante la fase de compilación y construcción (build). Aplicación que, como Maven (esta ya es aplicación), está pensada para gestionar procesos de automatizado de compilación.

► **JasperReports Designs:** asistente visual para la generación de informes (antiguo iReports) que forma parte de JasperSoft Studio.

► **JRXML:** formato XML resultante de implementar el informe (manualmente en Java por ejemplo o mediante JR Designs también). Es el código fuente del informe donde se recogen las particularidades del mismo.

► **JASPER:** formato binario nativo de JasperReports resultante de compilar con la herramienta **Ant** el fichero JRXML anterior. Al compilarlo, se pasa de un formato de **objeto JasperDesign** a un **objeto JasperReport**.

► **JRPRINT:** sin datos no hay paraíso. De ahí que haya que pasar de la carcasa del informe a un documento con los datos que se quiere presentar al usuario. Nuevo formato compilado pero ahora con los datos ya insertados. La salida generada por este proceso es un fichero .jrprint listo para visualizarse, imprimirse o exportarse a otros formatos como PDF, XML, HTML, CSV, XLS, RTF, TXT.

► **Sección de un reporte:** apartado diferenciado de un reporte en la que una vez ubicados datos, imágenes o letreros, permiten una

DESCRIPCIÓN DE LA TAREA A REALIZAR

0) Trabajo previo [disposición de los elementos necesarios y fundamentos]:

- Recupera el proyecto en Java realizado en la práctica anterior.
- Descarga [una versión actual y estable del entorno Jaspersoft Studio](#).
- Identifica qué herramienta de JasperReports suministra el asistente gráfico para generar informes.
- Averigua el motivo de que haya que compilar un reporte y las dos formas de hacerlo que hay.
- Elige Netbeans o Eclipse para realizar esta práctica.

1) Trabajo creativo [confección de una infografía (de cosecha propia aunque las imágenes puedan ser prestadas) con [Piktochart](#) o [Genially](#)]. En los tres apartados de la infografía deberán reflejarse:

- a) Cómo se integra la librería JasperReports en un proyecto Eclipse o NetBeans.
- b) Qué características diferencian a los tipos de informes incrustados y no incrustados.
- c)Cuál es la estructura de un reporte JasperReports, indicando las secciones que lo conforman.
- d) Cómo va el proceso (o ciclo de vida) de un informe así como las extensiones de los ficheros que se van dando en cada etapa.

2) Ahora que ya conocemos la teoría de cómo se genera un informe con JasperReports, nos remangamos para poner todo eso en práctica para plasmarlo en un ejercicio simple, una vez visto cómo lo hacen en los videos que se encuentran en [<https://youtu.be/ecaZnYpieqY>] y [<https://youtu.be/NeppFohXGMo>]:

- a) Implementa el reporte básico que muestra el primer vídeo tirando de una base de datos interna para pruebas que tiene el propio JasperStudio.
- b) Implementa una mejora del reporte básico anterior incluyendo un icono/logotipo en el título y un total final con la suma de los envíos realizados.
- c) Implementa una mejora del reporte básico anterior incluyendo un gráfico de barras que diferencie visualmente qué porcentajes presenta cada modo de envío (ShipVia).
- d) Obtén una salida PDF del reporte confeccionado en el apartado c).

3) Si queremos darle un punto mayor de profesionalidad a la práctica, parece lógico manejar datos que estén almacenados en BD. Por lo que se propone mejorar la aplicación Visor de forma que ahora los datos no estén en ArrayLists sino en una BD (MariaDB o MySQL vía JDBC). La diferencia con la ArrayList es que a la tabla se le añadirá un campo nuevo denominado Nacionalidad. A la aplicación se le deberá añadir dos botones para la visualización de sendos reporte que se implementarán a continuación.

4) A la vista de cómo nos ha quedado la aplicación Visor, vamos incorporarle reportes más complejos (datos agrupados y gráficos) una vez visto cómo lo hacen en los videos que se encuentran en [http://chuwiki.chuidiang.org/index.php?title=Ejemplo_b%C3%A1sico_con_Jasper_Report] y [<https://youtu.be/APghjrUJvYo>]:

- a) Confecciona un icono/logotipo que acompañará al título de ambos informes, apareciendo sólo al principio de los mismos.
- b) Implementa un reporte básico que presente todos los datos de la tabla de la BD del proyecto **Visor mejorado**, utilizando una de las plantillas que ofrece el asistente de JasperReports, de forma que se vean ordenados por fecha de apertura.
- c) Implementa una mejora del reporte básico anterior incluyendo subtotales de saldos por Nacionalidad (acumulado y promedio) así como un total final con el acumulado de todos los saldos.
- d) Implementa una mejora al reporte del apartado c) incluyendo un gráfico de tarta que diferencie visualmente qué porcentaje de clientes presentan un saldo mayor de 1000€ y quiénes no.
- e) Remata el principio de cada página con la fecha de ese momento así como cada pie de página con el número de página que le corresponde.
- f) Integra en el código Java o JavaFX (tú eliges) de la aplicación **Visor mejorado**, tanto el reporte básico como el reporte confección desde el punto c) que se presentarán tras el pulsado de los botones creados al efecto. EL primero se mostrará en PDF y el segundo en HTML. Si te vale de ayuda [<https://youtu.be/2DvwZmsHfgo>].

EVALUACIÓN

La calificación de cada una de las fases será:

Calificación de la práctica			
Apartado 1) a) 0.35 puntos b) 0.4 puntos c) 0.35 puntos d) 0.4 puntos	Apartado 2) a) 0.55 puntos b) 0.55 puntos c) 0.55 puntos d) 0.25 puntos	Apartado 3) 1 punto	Apartado 4) a) 0.25 puntos b) 1 punto c) 1.05 puntos d) 1.05 puntos e) 0.25 puntos f) 2 puntos

En la fecha estipulada, si el equipo no lo solicita con anterioridad, el profesor procederá a la revisión y calificación de la práctica junto con los alumnos cuya duración no deberá exceder los 15 minutos. Caso de suceder esto, restará de la nota final de la práctica. Una vez realizada dicha revisión, que requerirá de la presencia de ambos miembros del equipo que mostrarán los resultados de forma indistinta a petición del profesor, uno de los miembros (acordado por ellos previamente) subirá a los buzones correspondientes del aula virtual la información requerida (recogida en un vídeotutorial, cuyo enlace se entregará en formato TXT), siendo en este caso:



- a) Infografía del apartado 1.
- b) Salidas PDF resultantes del apartado 2.
- c) Código de la aplicación (Java / JavaFX) con la integración de los reportes.