**Desarrollo de Aplicaciones Multiplataforma**

***M06 – Acceso a Datos***

***Aplicación con Servlets***

Objetivos

* Programar componentes que gestionan información almacenada en ficheros.
* Programar componentes que gestionan, mediante conectores, información almacenada en bases de datos.
* Programar componentes que gestionan información utilizando mapeo objeto- relacional.
* Programar componentes que gestionan información almacenada en bases de datos objeto- relacional y orientado a objetos.
* Programar componentes que gestionan información almacenada en una base de datos nativa XML.
* Probar y documentar los componentes desarrollados.
* Integrar los componentes desarrollados en aplicaciones.

Actividad 5

***Aplicación con Servlets***

*Lee detenidamente todos los pasos a continuación. Realiza todas las acciones indicadas y responde a todas las preguntas planteadas.*

*Deberás realizar una aplicación* ***JAVA*** *que incluya todas las clases y métodos necesarios para dar respuesta a todos los ítems marcados en el enunciado.*

*La entrega de la actividad estará formada por el proyecto que hayas creado en* ***NetBeans*** *como solución de la actividad, comprimido en formato* ***.zip****.*

Deseamos construir una aplicación que permita acceder a los empleados registrados a sus incidencias en un sistema de gestión de incidencias.

Para ello, se plantea la necesidad de construir una infraestructura sobre una base de datos MySQL mediante JPA y EJB:

* Tabla **incidencias**: Albergará los datos de las incidencias. Contendrá para cada una la fecha-hora de creación, el origen, el destino, detalle de la incidencia y el tipo de la misma.
* Tabla **empleados**: Contendrá los datos de inicio de sesión de los empleados al sistema: nombre de usuario y contraseña. Además, incluirá los datos del empleado. Tendrá datos como el nombre completo y teléfono de contacto, como mínimo.
* Tabla **historial**: Registrará los eventos de inicio de sesión (I), consulta de incidencias destinadas para un empleado (C) y creación de incidencias urgentes (U). Para ello dispondrá de tres campos: el tipo de evento (“I”, “E” y “U” respectivamente), la fecha-hora del evento y el nombre de usuario del empleado que lo ha generado.

Con todo esto en mente, tenemos que realizar lo siguiente:

1. Construir los EJB’s que creamos necesarios sobre dichas tablas. ***(0.50 puntos)***
2. Crear un backend apropiado para los empleados. Llamar a la clase ‘**IncidenciasEJB**’. Mostrará un menú que deberá permitir: ***(3.00 puntos)***
   1. Insertar un empleado nuevo en la B.D.
   2. Validar la entrada de un empleado (suministrando usuario y contraseña)
   3. Modificar el perfil de un empleado existente.
   4. Cambiar la contraseña de un empleado existente.
   5. Eliminar un empleado.
3. Modificar el backend anterior de manera que gestione también las incidencias, muestre un menú con las siguientes opciones: ***(3.00 puntos)***
   1. Obtener un objeto Incidencia a partir de su Id.
   2. Obtener la lista de todas las incidencias.
   3. Insertar una incidencia a partir de un objeto de clase Incidencia definido adecuadamente según los campos que presenta (incluido el empleado que la origina y el empleado destino).
   4. Obtener las incidencias para un empleado a partir de un objeto de clase Empleado.
   5. Obtener las incidencias creadas por un empleado concreto.
4. Crear un backend apropiado para la gestión del historial. Llamar a la clase ‘**HistorialEJB**’. Mostrará un menú que deberá permitir: ***3.00 puntos)***
   1. Insertar los eventos mediante sendos métodos.
   2. Obtener la fecha-hora del último inicio de sesión para un empleado concreto.
   3. Obtener el ranking de los empleados por cantidad de incidencias urgentes creadas (más incidencias primero).
   4. Obtener la posición dentro del ranking para un empleado concreto.
5. Construir una aplicación Java mediante Servlets, que demuestre el uso de los diferentes elementos programados. ***(0.5 puntos)***

Este programa debe mostrar al usuario un menú con las opciones que sean necesarias para poder probar cada una de las funcionalidades.