Forma

Descripción generada automáticamente con confianza media

**PRÁCTICA 1 – RECOPILACIÓN, ESTRUCTURACIÓN Y ANÁLISIS DE DATOS**

SISTEMAS DE INFORMACIÓN

GRADO EN INGENIERÍA DE LA CIBERSEGURIDAD

CURSO 2024 – 25

Autores:

Alejandro Fuentes Contreras

Javier García Márquez

Juan Antonio Suárez Suárez

Contenido

[EJERCICIO 1 3](#_Toc192956622)

[EJERCICIO 2 6](#_Toc192956623)

[EJERCICIO 3 8](#_Toc192956624)

[EJERCICIO 4 9](#_Toc192956625)

# EJERCICIO 1

Esta parte del documento recoge el modelado del proceso de negocio para la gestión de incidencias de ciberseguridad en una empresa tecnológica. En concreto, se han realizado modelados en dos notaciones diferentes: *Business Process Modeling Notation* (BPMN) y *Unified Modeling Language* (UML).

El proceso de modelado comienza con la solicitud del gerente de operaciones de seguridad (SOC) a los analistas de seguridad para evaluar las incidencias recibidas. A partir de esta evaluación, se determina si se debe dar formación a los empleados, automatizar algún proceso o marcarlo para una revisión posterior.

La decisión tomada implicará a otros departamentos como recursos humanos y desarrollo, quienes implementarán las soluciones correspondientes.

Por último, se realiza un seguimiento de las mejoras para evaluar su impacto en la organización.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

Ilustración 1. Modelado de proceso UML

Como se puede observar en el diagrama UML, una vez bifurcado el flujo en el gerente de operaciones, el departamento de RRHH realiza sus acciones y, una vez notificados los empleados, termina su flujo, es decir, a partir de ahí ya no se realizan más tareas que afecten al flujo general.

Al inicio de la actividad del departamento de desarrollo nos encontramos una región interrumpible. En este caso, si el coste de la automatización supera lo 10.000 € habrá que consultar con el gerente de operaciones, en otro caso, se continuará con su desarrollo.

Por último, todo el flujo viene a converger en un punto de espera de 1 mes, después del cual se valuará la efectividad de las medidas tomadas y pondremos final al flujo.

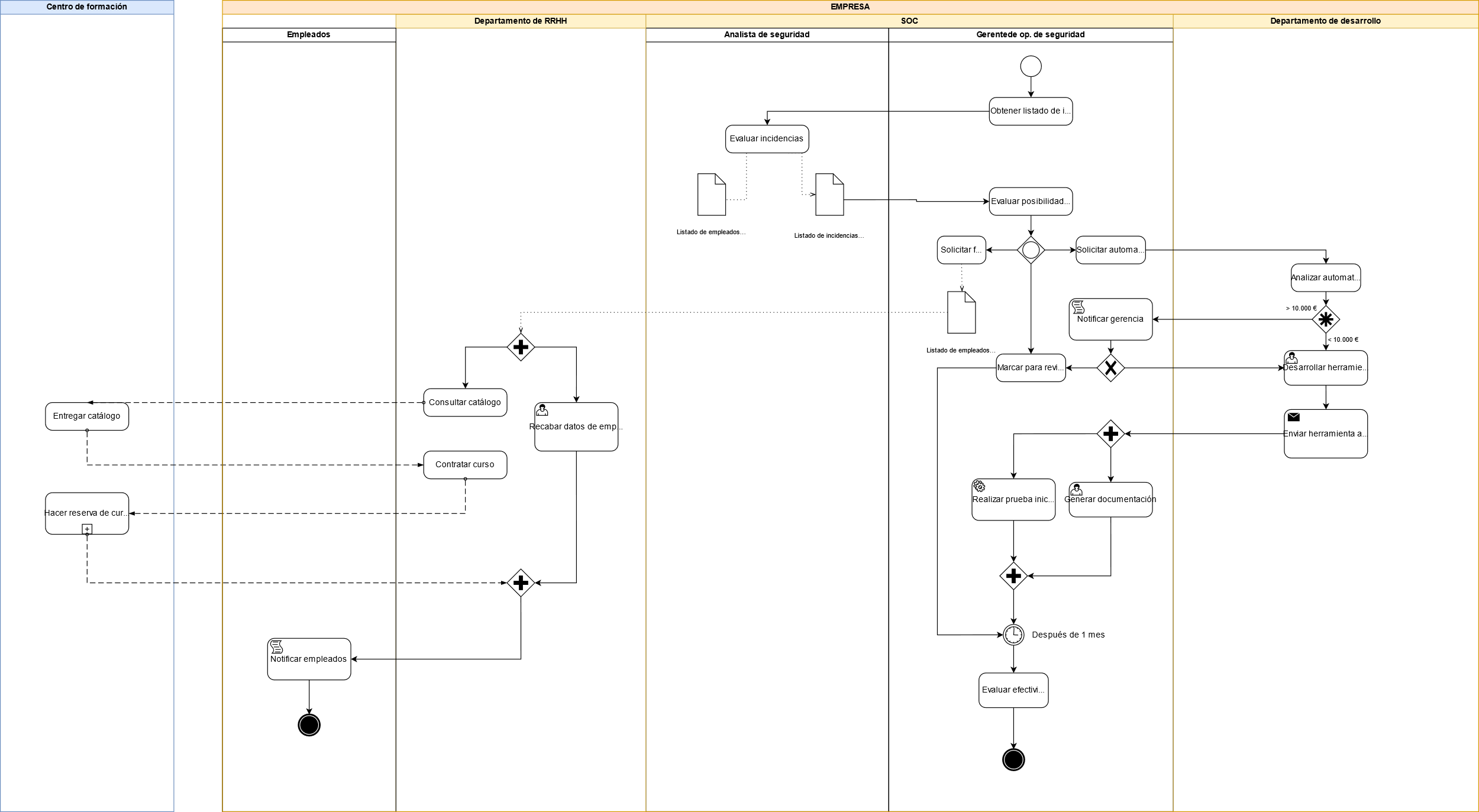


Ilustración 2. Modelado de proceso BPMN

Como se puede observar el diagrama BPMN nos ofrece mucha más flexibilidad y posibilidades de representar de manera más correcta parte del flujo.

Lo que podemos destacar del diagrama es la adición de dos actores más, los empleados y el centro de formación. Este último tienen su *pool* separado del resto pues en BPMN los *pools* pertenecientes a una misma organización estarán juntos, mientras que los que sean externos estarán en otro diferente.

Ahora, el flujo de RRHH intervendrá con los empleados y con el centro de formación.

Las distintas actividades utilizadas son:

|  |  |  |
| --- | --- | --- |
| SCRIPT | Texto, Icono  El contenido generado por IA puede ser incorrecto. | Esta tarea es automatizada, por ello el uso de esta actividad. |
| SUB-PROCESS | Texto, Icono  El contenido generado por IA puede ser incorrecto. | Esta tarea muestra una tarea que debería dividirse en más subprocesos. |
| USER | Icono  El contenido generado por IA puede ser incorrecto. | Esta actividad requiere la intervención de un usuario para su realización. |
| SERVICE | Icono  El contenido generado por IA puede ser incorrecto. | Esta tarea es ejecutada automáticamente por un sistema externo o un servicio sin intervención humana. |
| SEND | Icono  El contenido generado por IA puede ser incorrecto. | Esta actividad envía un mensaje a otro sistema o entidad. |

En este caso, en vez de una región de interrupción, tenemos una bifurcación “compleja”, es decir, según las condiciones que se den (si el presupuesto es mayor o menor de 10.000 €) se ramificará hacia un lado u otro. Además, si el presupuesto se supera y pasa a revisión del gerente de operaciones, este deberá decidir si se continúa con el desarrollo o se marca para una revisión posterior, de ahí la bifurcación ‘*exclusive’*, que sólo permitirá tomar un flujo.

Los dos diagramas anteriormente presentados se encuentran subidos en el repositorio de GitHub de esta práctica: <https://github.com/juansrz/SI-Practica1-Grupo18/tree/main>

# EJERCICIO 2

En esta parte de la práctica, se implementará un sencillo sistema ETL que nos permita estructurar y analizar los datos provenientes de un archivo JSONque se nos proporciona, almacenarlos en una base de datos SQLite y, posteriormente, facilitar su consulta para la toma de decisiones en la gestión de incidencias.

Todo el código se puede consultar en el siguiente repositorio de GitHub: <https://github.com/juansrz/SI-Practica1-Grupo18/tree/main>

En primer lugar, en el archivo main.py, comenzaremos con la **extracción y transformación de datos**.

El código comienza con la creación de la base de datos mediante SQLite, se crea la tabla y las filas y columnas necesarias para la representación de los datos de nuestro JSON.

Después procedemos a la apertura del archivo “datos.json”, que contiene la información sobre clientes, empleados, tipos de incidentes y tickets emitidos. A continuación, utilizamos json.load() para cargar los datos en memoria en forma de diccionario.

Cada conjunto de datos dentro del JSON (clientes, empleados, tickets, etc.) se almacena en listas dentro del diccionario.

La carga de datos sigue un orden estratégico para garantizar la integridad:

1. Eliminamos las tablas preexistentes con DROP TABLE IF EXISTS para evitar conflictos de posibles datos obsoletos.
2. Creamos las tablas con CREATE TABLE, asegurándonos de que las relaciones entre ellas están correctamente definidas mediante claves primarias y foráneas.
3. Insertamos los datos en las tablas correspondientes mediante INSERT INTO.

Por último, para validar la carga de datos, usaremos Pandas para leer y mostrar información básica:

* pd.read\_sql\_query("SELECT \* FROM tickets;", conn): Obtiene todos los tickets registrados.
* len(df\_tickets): Muestra el número total de incidencias registradas.
* pd.read\_sql\_query("SELECT \* FROM contactos\_empleados;", conn): Permite analizar qué empleados han participado en la solución de incidencias.

Para continuar con el ejercicio, pasamos al archivo analisis.py, donde se calculan los siguientes valores con sus respectivos resultados:

|  |  |
| --- | --- |
| **Cálculo** | **Resultado** |
| Numero de muestras totales. |  |
| Media y desviación estándar del total de incidentes en los que ha habido una valoración mayor o igual a 5 por parte del cliente. |  |
| Media y desviación estándar del total del número de incidentes por cliente. |  |
| Media y desviación estándar del número de horas totales realizadas en cada incidente. |  |
| Valor mínimo y valor máximo del total de horas realizadas por los empleados. |  |
| Valor mínimo y valor máximo del tiempo entre apertura y cierre de incidente. |  |
| Valor mínimo y valor máximo del número de incidentes atendidos por cada empleado. |  |

Tabla 1. Resultados cálculos ejercicio 2

# EJERCICIO 3

# EJERCICIO 4