

GEST-AAS

Alfredo de los Rios Carrizo

Trabajo Fin de Grado Superior de 2º de
Desarrollo de Aplicaciones Multiplataforma
<https://github.com/AlfredodlRC/Gest-aas.git>

Tutor: Raquel Cerdá

Instituto Tecnológico

Edix

2023/2024

1. Introducción.

El proyecto se ha desarrollado con el objetivo de que los usuarios de esté puedan tener un catálogo de activos, de amenazas y de salvaguardas de la organización a la que pertenece. Este catálogo es especialmente útil cuando estamos hablando del tema del análisis de riesgos dentro de la seguridad tanto informática como de la información, puesto que permite saber cual es el estado de la organización en esté aspecto y consecuentemente poder tomar las decisiones adecuadas en cada momento.

Dentro del mercado hay programas que te permiten realizar inventarios, pero estos generalmente se limitan a los equipos informáticos y/o de red, sin tener en cuenta otros elementos que pueden influir en la seguridad como pueden ser las personas o las instalaciones, así como los propios servicios que se prestan.

El programa que más se ajusta a los objetivos del proyecto sería el programa pilar :

<https://pilar.ccn-cert.cni.es/index.php>.

Para desarrollar el proyecto se ha basado en la metodología magerit elaborada por el antiguo Consejo superior de administración electrónica, que está basada en el standat ISO27000.

https://administracionelectronica.gob.es/pae_Home/pae_Documentacion/pae_Metodolog/pae_Magerit.html

2. Palabras clave

Activo: Recursos del sistema de información o relacionados con éste, necesarios para que la Organización funcione correctamente y alcance los objetivos propuestos por su dirección.

Activo superior: (respecto a otro) Activo el cual necesita el otro activo para realizar su función

Activo inferior: (respecto a otro) Activo que el otro activo necesita para realizar su función

Nivel de valoración: Situación dentro de una escala del valor de un activo.

Valor económico: es el precio que se asigna a un activo

Valor acumulado: es el valor teniendo en cuenta el valor de los activos inferiores y su grado de dependencia con el activo.

Valor repercutido: es el valor teniendo en cuenta el valor de los activos superiores y su grado de dependencia con el activo.

Amenaza: causa potencial de un incidente que puede causar daños a un sistema de información o a una organización.

Degradación: Pérdida de valor de un activo como consecuencia de la materialización de una amenaza.

Probabilidad: posibilidad de que algo (una amenaza) ocurra

Impacto: Consecuencia que tiene la materialización de una amenaza sobre un activo

Riesgo: Estimación del grado de exposición a que una amenaza se materialice sobre uno o más activos causando daños o perjuicios a la Organización.

Salvaguarda: Procedimiento o mecanismo tecnológico que reduce el riesgo.

Eficiencia: grado de efectividad de una salvaguarda sobre una amenaza

Degradación residual: Degradación de una amenaza teniendo en cuenta un paquete de salvaguardas.

Probabilidad residual: probabilidad teniendo en cuenta un paquete de salvaguardas.

Impacto residual: impacto teniendo en cuenta un paquete de salvaguardas.

Riesgo residual: riesgo teniendo en cuenta un paquete de salvaguardas.

3. Índice general.

Índice de contenido

1. Introducción.....	2
2. Palabras clave.....	3
3. Índice general.....	4
4. Módulos formativos aplicados en el trabajo.....	5
5. Herramientas/Lenguajes utilizados.....	6
5.1 Herramientas.....	6
Eclipse IDE for java developers.....	6
MySQL Workbench 8.0.....	6
STARUML.....	6
Git/GITHUB.....	7
5.2 Lenguajes.....	7
JAVA.....	7
SQL.....	7
6. Componentes del equipo.....	8
7. Fases del proyecto.....	9
7.1 Estudio de mercado.....	10
7.2 Modelo de datos utilizado.....	11
7.3 Diagramas UML.....	12
7.4 Diseño de las interfaces.....	13
7.5 Planificación del desarrollo.....	14
7.6 Explicaciones de la funcionalidad del proyecto.....	15
9. Conclusiones y mejoras del proyecto.....	16
Bibliografía.....	17
11. Anexos.	18

4. Módulos formativos aplicados en el trabajo.

Base de datos

La aplicación utiliza una base de datos para almacenar los activos, las amenazas y las salvaguardas. Además de las tablas de estos elementos y sus tablas tipos se implementará diversas tablas para determinar la relación e interacción entre los diferentes tipos de elementos.

Programación

Esta asignatura es básica para el desarrollo del proyecto, ya que para realizar la aplicación se utilizará la programación orientada a objetos explicada en esta asignatura.

Acceso a datos

Al utilizar las dos anteriores asignaturas es, casi, obligatorio el uso de esta asignatura sobre todo teniendo en cuenta que la aplicación hace uso de la base de datos.

Desarrollo de interfaces

La aplicación está concebida como una aplicación de escritorio. Teniendo en cuenta que la mayoría de sistemas operativos actuales la interacción con el usuario utilizan sistemas GUI, se decidió utilizar un sistema de ventana para la interacción con el usuario.

Entorno de desarrollo

Para desarrollar la aplicación se ha tenido que utilizar varias herramientas de desarrollo. También se ha utilizado un sistema de versión, tema que se trató en esta asignatura. Por último se ha desarrollado varios diagramas que también se ven.

Lenguaje de marcas

En principio se tenía previsto hacer una importación/exportación de elementos mediante xml utilizando los esquemas XSD que existen en la documentación de Magerit, no llegando a realizarse finalmente.

5. Herramientas/Lenguajes utilizados.

5.1 Herramientas

Eclipse IDE for java developers (<https://www.eclipse.org/ide/>)

Eclipse es un IDE compuesto por un conjunto de herramientas de programación de código abierto que permiten trabajar en diferentes lenguajes tanto de programación como de marcas. Originalmente fue desarrollado por IBM, pero actualmente la Fundación Eclipse, es el responsable de su desarrollo. Se ha decidido incorporar para el proyecto el proyecto el PlugIns amateras para poder extraer las clases construidas

Vídeo explicativo del Plugin amateras

https://www.youtube.com/watch?v=6ha62S8E_5I&list=PLnNbmcjjevxsAT5ZOUvcR5BFczXcQj09M&index=3

Repositorio del Plugin amateras

<https://github.com/takezoe/amateras-modeler>

MySQL Workbench 8.0 (<https://dev.mysql.com/downloads/workbench/>)

Herramienta de estilo visual para el trabajo con bases de datos MySQL

Apache OpenOffice (<https://www.openoffice.org/es/>)

Suite ofimática inicialmente open source actualmente es propiedad de Sun Microsystem. Es una competidora multiplataforma directa de Office Microsoft, poseyendo diferentes desarrollos derivados como LibreOffice

STARUML (<https://staruml.io>)

Herramienta visual de MKLab cuya principal funcionalidad es desarrollar diagramas UML.

Git/GITHUB (<https://git-scm.com>, <https://github.com>)

Git es un sistema de gestión de versiones diseñado por Linus Torvalds. El propósito del sistema es llevar un control sobre las diferentes versiones del código fuente de proyectos a partir de cierto tamaño.

Github es un sitio web que permite el almacenaje y gestión de proyectos utilizando Git.

5.2 Lenguajes

JAVA

Lenguaje de programación multiplataforma. Se ha elegido este lenguaje de programación principal por dos motivos, el primero porque ha sido el lenguaje troncal de la formación en DAM y segundo porque al ser un lenguaje multiplataforma el programa producido puede ser ejecutado en cualquier equipo que ejecute una máquina virtual JVM.

SQL

Lenguaje de programación destinado al acceso de base de datos relacionales. En la práctica es un lenguaje estandarizado para trabajar con las bases de datos relacionales.

6. Componentes del equipo

Alfredo de los Rios Carrizo

7.Fases del proyecto

Las fases del proyecto son las típicas de un proyecto,

- Inicio del proyecto, definir que se pretende realizar el proyecto, en este, la gestión de activos, amenazas y salvaguardas.
- Definición del proyecto, declarar como se va a desarrollar el proyecto, inicialmente de creará la base de datos y la GUI de la aplicación y posteriormente se procederá a dar las diferentes funcionalidades a la aplicación.
- Desarrollo del proyecto, ir realizando los diferentes pasos para realizar el proyecto
- Supervisión del proyecto, evaluar y realizar las correcciones correspondiente sobre el proyecto
- Cierre del proyecto

El proyecto completo sobrepasará completamente el tiempo destinado el tiempo de dedicación a este trabajo de fin de curso, por lo que se intentará dar una versión con las funcionalidades necesarias para que el tutor y el equipo de evaluación del trabajo tenga una idea del proyecto total a desarrollar.

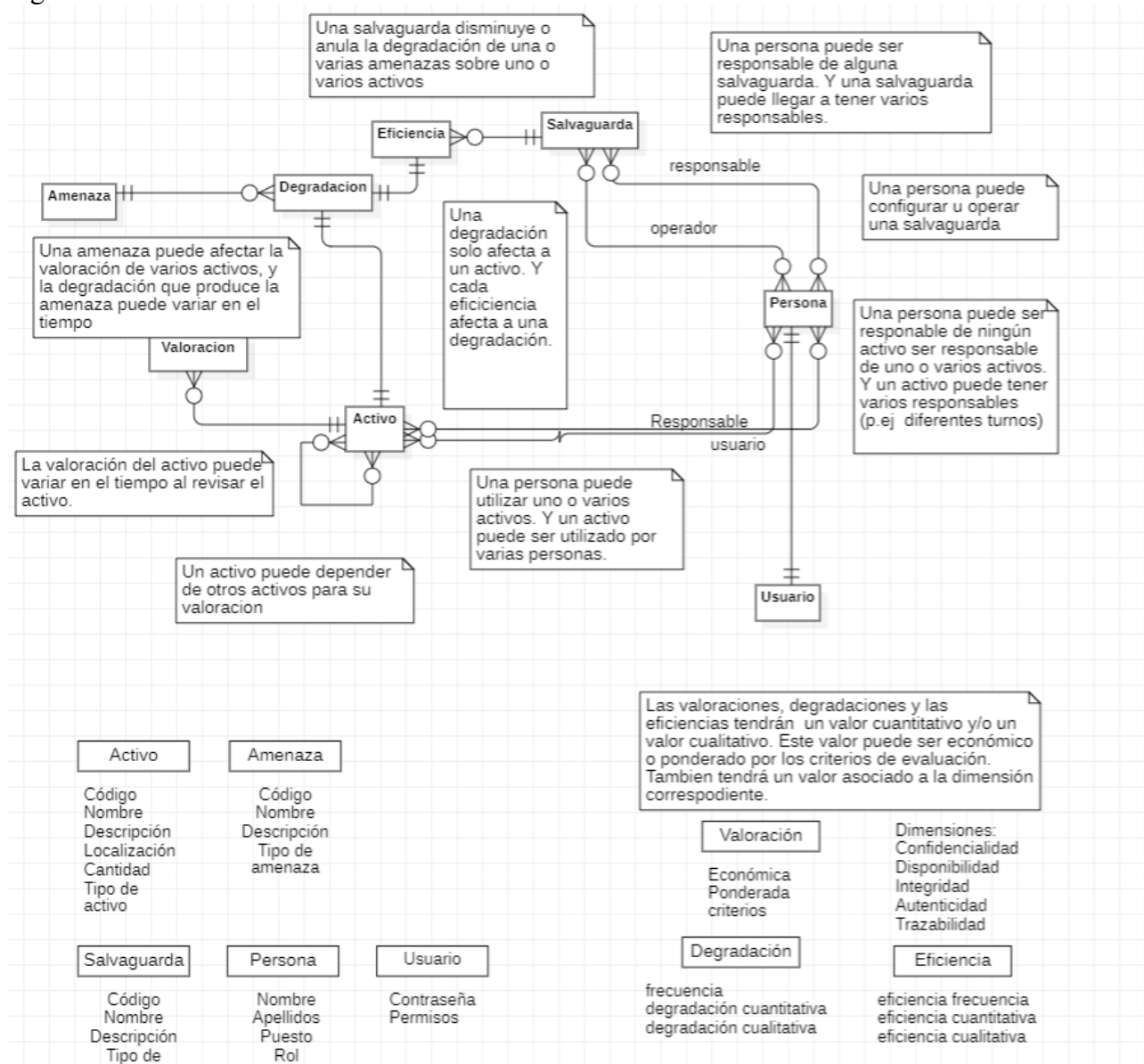
7.1 Estudio de mercado

Investigando sobre el mercado del software de gestión de la seguridad informática, el desarrollador se percató de que la mayoría de los programas desarrollados sobre el tema o se centraban o en gestión de vulnerabilidades o se centraban en los activos estrictamente tecnológicos, de tal forma que los primeros se centraban cuando ocurren alguna incidencia y los segundos no cubren la evaluación de los elementos no informáticos como pueden ser las personas, las instalaciones o los servicios en si mismos.

Se ha encontrado una excepción notable como es el caso del programa pilar (<https://pilar.ccn-cert.cni.es/index.php>), el cual si se acerca a los objetivos de este proyecto.

7.2 Modelo de datos utilizado

Para la realización del modelo de datos de la base de datos se ha partido del siguiente esquema lógico:



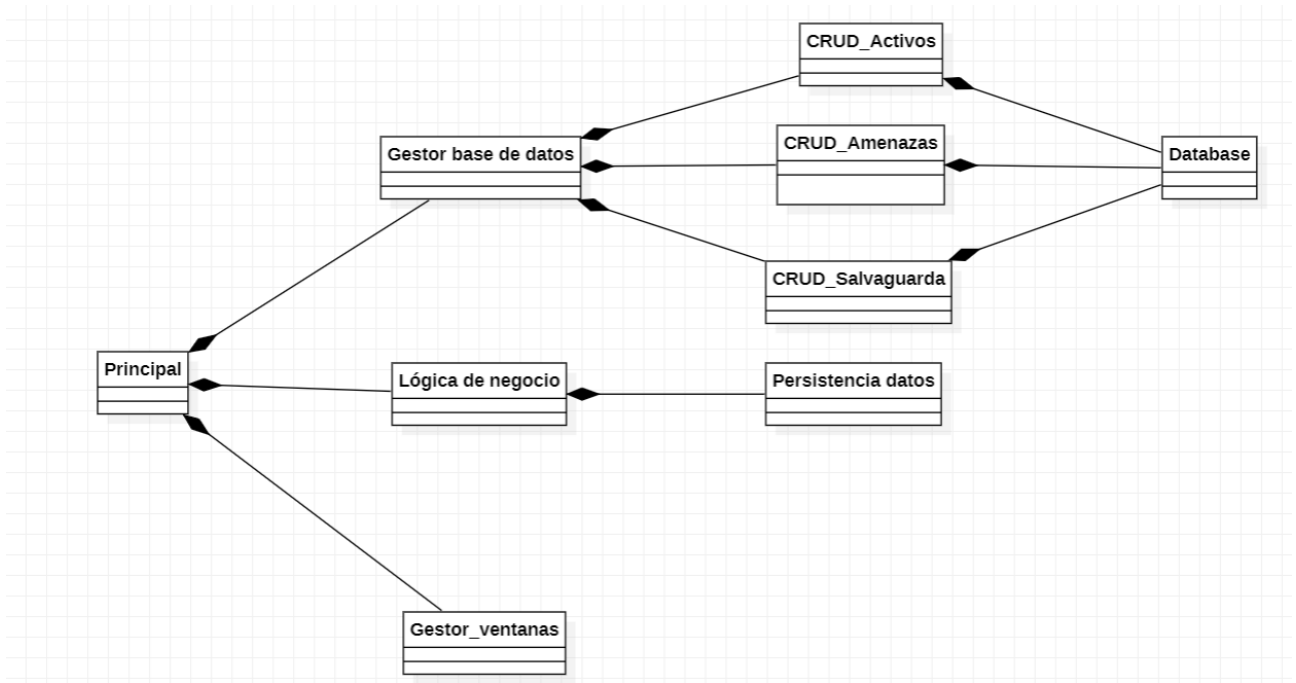
(entidad-relacion bbdd.png en el directorio documentacion/diagramas)

Después de analizar el grafo anterior y aplicando las diversas funciones de normalización las tablas de la base de datos son las siguientes:

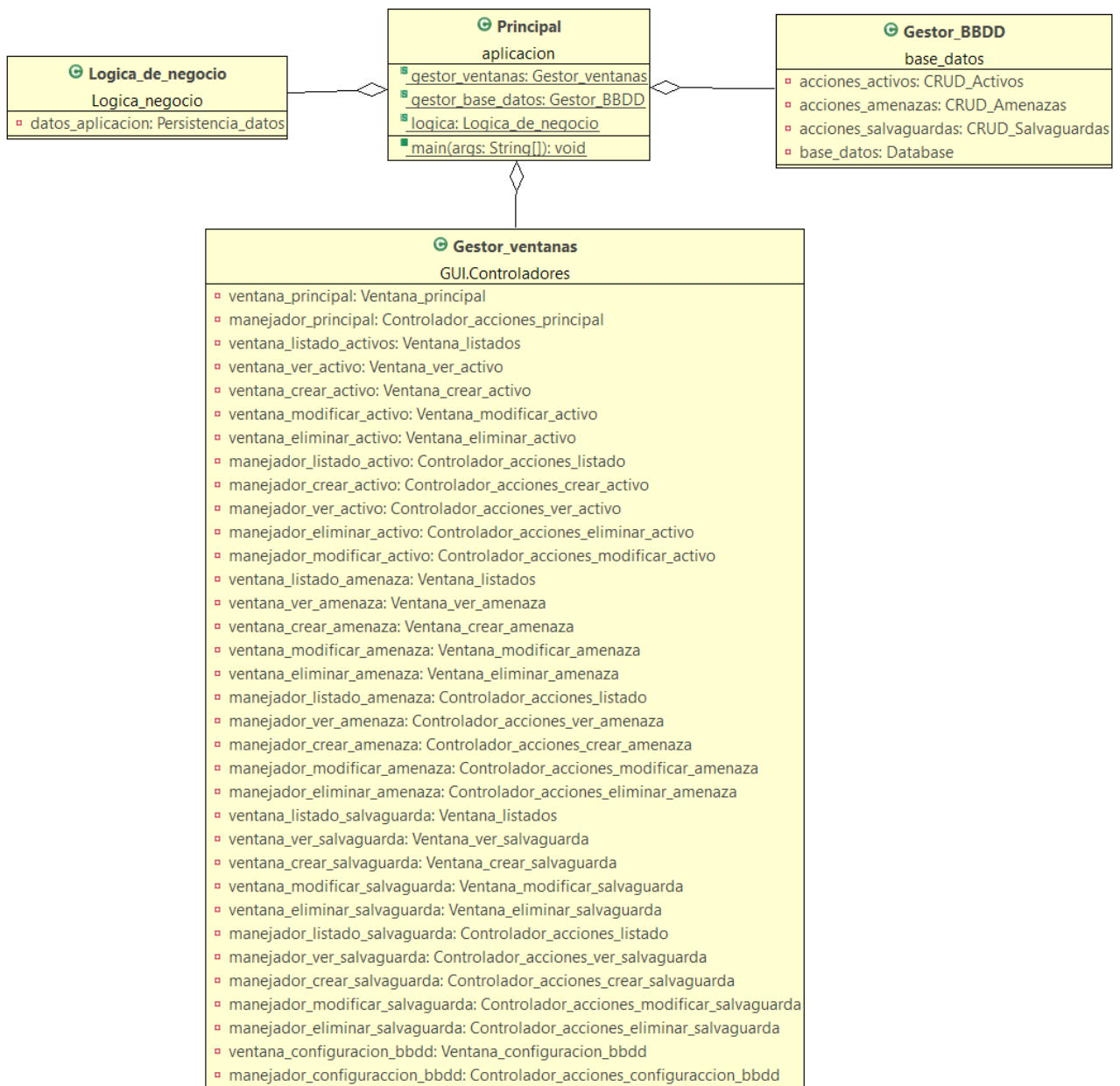
Como se puede observar la base de datos está incompleta, sin rematar, esto es debido a que la funcionalidad relacionadas con las escalas de impacto, de riesgo y de probabilidad con las tablas de degradación y de eficiencia se desarrollará en una fase posterior. Por otra parte la tabla de persona debería de estar relacionada tanto con la tabla de activo, mediante dos relaciones una como responsable del activo y otra como usuario que utiliza ese activo, como con la tabla de salvaguardas, también con las dos relaciones especificadas para la tabla de activos.

7.3 Diagramas UML

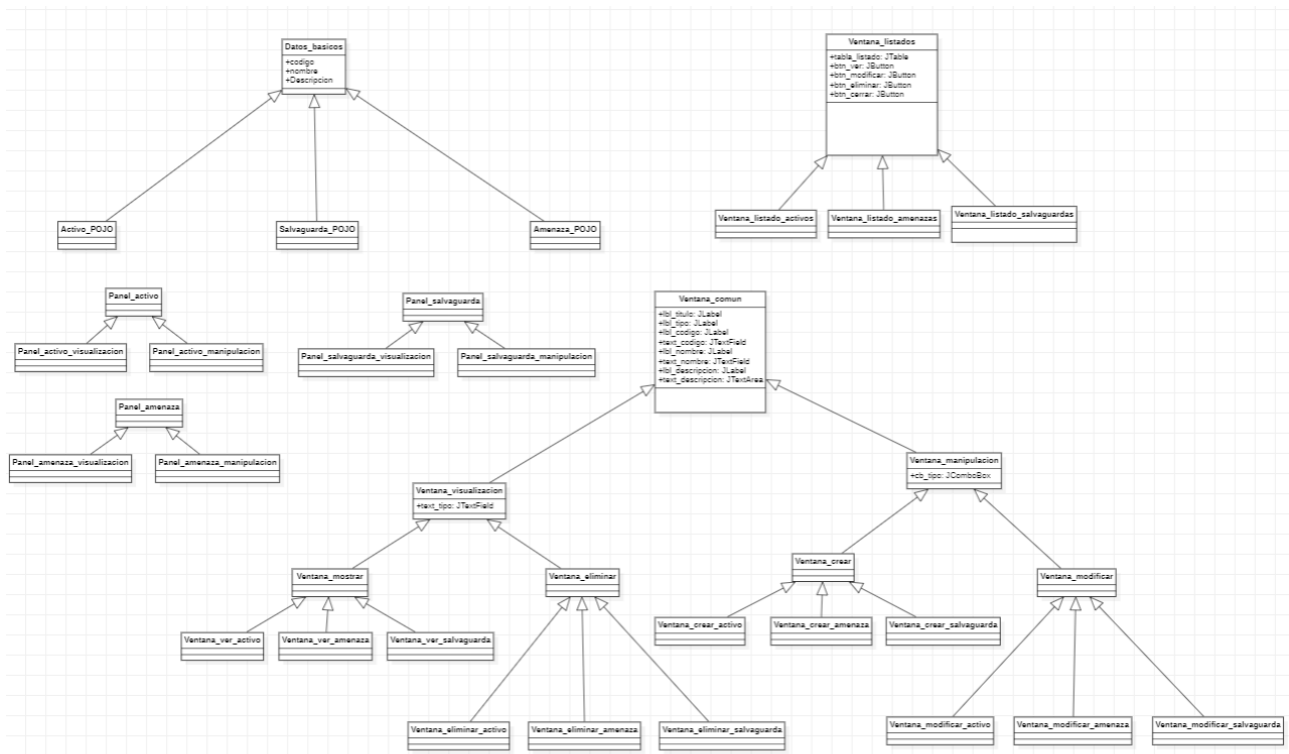
Inicialmente la estructura inicial de la aplicación fue a siguiente:



Quedando esta estructura finalmente:



Y se definición el sistema de clases a desarrollar como se muestra en la siguiente imagen:



Los diagramas que muestran el estado actual de las clases de la aplicación se encuentra en el directorio Documentacion\Diagramas del repositorio del proyecto.

7.4 Diseño de las interfaces

La pantalla principal tiene este diseño inicial :



La estructura general de las pantallas que trabajan con los elementos es la siguiente:

Tipo

Descripción

Código

Nombre

Fecha

Panel datos especificos

Las pantallas de los listados de los elementos siguen la siguiente estructura:

Listado de elemento

Diagram illustrating the structure of the 'Listado de elemento' window. It consists of a large light blue rectangular area labeled 'Listado de elementos'.

Cerrar

Estructura de la ventana de configuración de la base de datos

Diagram illustrating the structure of the database configuration window. It contains the following fields and controls:

- host
- puerto
- nombre bbdd
- usuario
- contraseña
- contraseña repetida
- Cancelar
- Aceptar

7.5 Planificación del desarrollo

El proyecto se planifico a partir de la creación de la base de datos y de desarrollo del sistema de pantallas, para que a partir de esté punto de partida y a través de un modelo incremental para dar cada vez más funcionalidad a la aplicación. De está manera se asegura tener algo ya operativo a la hora de presentar el proyecto.

Al realizar el diseño de las diferentes ventanas se vio que contenían algunos elementos comunes con lo se pensó en utilizar la herencia para desarrollar estas partes comunes.

Finalmente la temporalización del proyecto fue la siguiente:

- Semana 2/10-8/10

* Definición del proyecto

* Presentación inicial del proyecto

- Semana 9/10-15/10

* Creación del modelo entidad-relación de la base de datos

* Creación de la pantalla principal sin implementar las operaciones

- Semana 16/10-22/10

* Implementación de la base de datos

* Diseño inicial de las pantalla

- Semana 23/10-29/10

* Implementación de los elementos comunes de las pantallas

- Semana 30/10-5/11

* Implementación del acceso a la base de datos

* Re diseño de la estructura general del programa

* Realización de la presentación intermedia

- Semana 6/11-12/11

* Implementación de las funciones especificas de las pantallas

- Semana 13/11-19/11

* Implementación de las funciones especificas de las pantallas

- Semana 20/11-26/11

* Implementación específicas de las pantallas de las pantallas

- Semana 27/11-3/12

* Implementación de las funciones específicas de las pantallas

- Semana 4/12-10/12

* Comprobación del proyecto y realización de la documentación a presentar

- Semana 11/12-17/12

* Presentación del proyecto

Para concretar los pasos seguidos se muestra el resultado del repositorio donde se indica las acciones generales realizadas en el proyecto-

```
G:\DAM\Gest-aas>git log --graph --oneline
* 8021617 (HEAD -> main, origin/main, origin/HEAD) ventana modificar activo funcional con algún fallo
* 5b3311d ventanas ver y eliminar de amenazas y eliminar de salvaguardas operan con BBDD
* 26daef2 ventanas ver y eliminar activos totalmente operativas
* 3c95507 agregada sistema de eficiencias y degradaciones en crear y modificar amenaza
* 0329038 pantalla configuracion BBDD implementada
* 6e692e0 implementado las acciones de las salvaguardas y probar conexion
* c398d61 eliminar amenaza implementado
* 381bd84 crear amenaza y modificar amenazas implementados
* 609a1a6 Datos incorporados a ver amenaza
* dbeea2d pantalla crear activo terminada y ver amenaza operativa sin datos
* e820c22 Añadida funcionabilidad a las ventanas de activos
* cee260a Cargar de criterios y escalas desde la BBDD
* 42b0b78 gestor de ventanas incorporado, inicializado funcionalidad crear activo
* c195e0e ventanas activos impementadas sin funcionabilidad
* 79995e1 BBDD creada
* 893339d Ventana principal - esquemas iniciales
* cf0f6cc Archivos base iniciales
* 5ae6e82 Initial commit
```

7.6 Explicaciones de la funcionalidad del proyecto.

La aplicación posee una ventana principal la cual sirve para ver los 3 listados, el de activos, el de amenazas y el de salvaguardas. Los cuales servirán para seleccionar el elemento sobre el que realizar las operaciones de ver, modificar y eliminar. Esta ventana posee un menú para poder realizar las diferentes operaciones. Algunas de estas funciones como la de importar del menú archivo y todas del menú usuario no se ha desarrollado todavía.

Las ventanas de listados permiten ver una tabla con los datos básicos de los elementos, además de unos datos adicionales (aún no se ha hecho las operaciones para mostrar estos datos).

En el panel de los activos posee las siguientes pestañas:

- Valor: En esta pestaña se muestra/determina el valor del activo y su nivel de valoración
- Criterios: Esta pestaña muestra/determina la valoración según diferentes criterios
- Activos superiores: Esta pestaña o muestra la lista superiores del activo o permite seleccionar o deseleccionar los activos no relacionados con el activo como activos superiores y determinar el grado de dependencia.
- Activos inferiores: Esta pestaña o muestra la lista inferiores del activo o permite seleccionar o deseleccionar los activos no relacionados con el activo como activos inferiores y determinar el grado de dependencia.

En el panel de las amenazas posee las siguientes pestañas:

- Activos: Esta pestaña permite ver los activos relacionados con la amenaza en las ventanas de ver y de eliminar, en las ventanas de crear y modificar se puede seleccionar los activos que afectan la amenaza y determinar la degradación que produce sobre los activos.
- Salvaguardas: esta pestaña muestra las salvaguardas que afectan a la amenaza correspondiente pudiendo agregar o eliminar esta relación.

En el panel de las salvaguardas posee las siguientes pestañas:

- activos: Aún no implementada, esta pestaña debería mostrar/manipular la relación de las salvaguardas con los activos y las amenazas que le afectan.
- salvaguardas: Aún no implementada, esta pestaña debería mostrar/manipular la relación de las salvaguardas con las amenazas y los activos a los que afectan.

9. Conclusiones y mejoras del proyecto

Conclusiones

Las conclusiones que se sacan del proyecto son bastantes positivas, aunque no se haya realizado la parte de asignar las salvaguardas ni a amenazas ni a activos, esta funcionalidad se puede realizar creando primero la salvaguarda y luego modificando las amenazas que vayan a ser afectadas por la salvaguarda.

Otra conclusión que se ha sacado es que el proyecto propuesto era demasiado amplio para el trabajo fin de curso. Teniendo en contrapartida, que servirá como punto de partida para desarrollar el proyecto de una manera de una forma más eficiente y detallista.

Mejoras

Las mejoras a realizar en el proyecto serán las siguientes:

- Completar la base de datos relacionando la tabla de los usuarios con las tablas de activos y de salvaguardas.
- Rellenar con los datos pertinentes la tabla `rel_tipo_activo_tipo_amenaza`.
- Completar la función de traer activos por tipo de amenaza para que tenga en cuenta la tabla del punto anterior.
- Completar las ventanas de las salvaguardas para que estas sean completamente operativas.
- Revisar concienzudamente la aplicación para que no tenga fallos.
- Mejorar el aspecto gráfico de la aplicación para ofrecer una mejor operatividad y una buena experiencia de usuario.
- Implementar un sistema de usuario de la aplicación y determinar que acciones puede hacer cada usuario según su rol.
- Implementar la importación/exportación de elementos mediante ficheros xml que cumplan con los esquemas XSD que existen en el magerit.
- Realizar un sistema de logs de la actividad realizada con la aplicación.
- Implementar el almacenaje del estado de la aplicación al cerrar y de los datos de la conexión a la base de datos.
- Implementar en el inicio del programa una opción para conectar a una base de datos en caso de que la aplicación no se conecte a la base de datos.

10.Bibliografía.

Las referencias básicas para realizar el proyecto han sido los apuntes y la documentación perteneciente al curso DAM. Además para realiza el proyecto se ha consultado el siguiente material en la red:

Magerit

https://administracionelectronica.gob.es/pae_Home/pae_Documentacion/pae_Metodolog/pae_Magerit.html

Referencia general del objetivo del proyecto

<https://pilar.ccn-cert.cni.es/index.php/metodologia/implementacion>

Referencia para markdown

<https://www.markdownguide.org>

Duplicar tablas

<https://trucosyayudas.com/copiar-y-crear-una-tabla-mysql-con-una-select/>

Control swing para horas

https://codigo--java.blogspot.com/2014/10/uso-basico-de-elementos-swing-con_8.html

Tabbed panel

<https://docs.oracle.com/javase/tutorial/uiswing/components/tabbedpane.html>

JTable y Default TableModel

<https://www.tutorialspoint.com/how-to-create-defaulttablemodel-which-is-an-implementation-of-tablemodel>

Poner cabecera al JTable - última respuesta

https://foro.elhacker.net/java/jtable_no_muestra_los_nombres_de_cada_una_de_las_columnas-t442411.0.html

Video para generar diagramas de clases a partir del código Java

https://www.youtube.com/watch?v=6ha62S8E_5I&list=PLnNbmcjjevxsAT5ZOUvcR5BFczXcQj09M&index=3

Repositorio para el plugins Amaretas

<https://github.com/takezoe/amateras-modeler>

Instalar XJC

<https://javaee.github.io/jaxb-v2/>

Referencia ejecutar xjc

https://docs.oracle.com/cd/E17802_01/webservices/webservices/docs/1.6/jaxb/xjc.html

11. Anexos.

Para crear la base de datos se podrá utilizar el archivo 'bbdd final.sql' situado en el directorio BBDD del repositorio. De momento para acceder a la base que se cree para probar la aplicación hay que modificar los datos contenidos en la clase contenida en el archivo '/app/src/datos/POJOS/Database_pojo.java'.