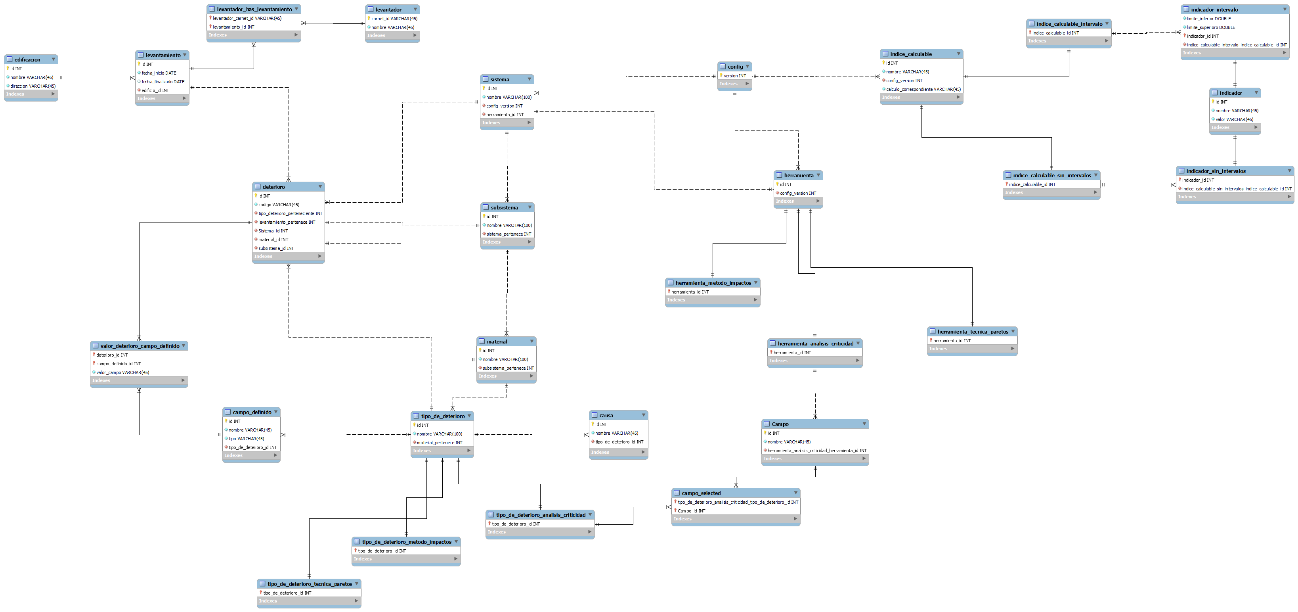
**Backend**

**Modelo de datos**



El modelo de datos es el cimiento sobre el cual se construye la estructura de almacenamiento y recuperación de información del sistema. Representa la esencia misma de cómo se organizan, relacionan y persisten los datos que fluyen a través de la aplicación. En este capítulo, se explorará las profundidades de este modelo, desde las tablas de la base de datos hasta las relaciones entre entidades.

**Entidades y sus relaciones**

|  |  |
| --- | --- |
| **Edificación** | |
| **Descripción Entidad**: Esta entidad representa las distintas edificaciones registradas. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna a la edificación en el momento de registrarla |
| nombre | Representa el nombre de la Edificación, por ejemplo: (Hospital Frank País) |
| Dirección | Representa la dirección donde se encuentra ubicada dicha edificación |

Tabla 1

**Relaciones:**

**Relación de 1-m** entre las entidades **Edificación** y **Levantamiento**: Esta relación indica que una edificación tiene muchos levantamientos registrados, y un levantamiento pertenece a una única edificación.

|  |  |
| --- | --- |
| **Levantamiento** | |
| **Descripción Entidad**: Esta entidad representa los distintos levantamientos realizados a las Edificaciones. Un levantamiento dentro del contexto de la evaluación del estado técnico de una edificación se refiere a la evaluación detallada de la estructura y los componentes de la Edificación. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al levantamiento en el momento de registrarlo |
| fecha\_inicio | Representa la fecha en la que se comienza a realizar el levantamiento de una edificación |
| fecha\_finalizado | Representa la fecha en el que se finaliza el levantamiento (Esta fecha está relacionada con el momento clave y culminante de finalizar el trabajo realizado de inspección en ese levantamiento) |
| edificación\_id | Este atributo identifica la Edificación a la cual pertenece dicho levantamiento (Atributo propio de una relación de 1 a muchos) |

Tabla 2

**Relaciones:**

**Relación de m-m** entre las entidades **Levantamiento** y **Levantador**: Esta relación indica que un levantamiento es realizado por muchos levantadores, y un levantador puede formar parte de varios levantamientos.

**Relación de 1-m** entre las entidades **Levantamiento** y **Deterioro**: Esta relación indica que un levantamiento tiene muchos deterioros identificados, y un deterioro es identificado en un único levantamiento.

|  |  |
| --- | --- |
| **Levantador** | |
| **Descripción Entidad**: Esta entidad representa las distintas edificaciones registradas. | |
| **Atributo** | **Descripción Atributo** |
| carnet\_id | Este atributo representa el identificador único de cada levantador dentro de la base de datos |
| nombre | Representa el nombre completo del levantador incluyendo segundos nombre y apellidos |

Tabla 3

Dentro del contexto de los levantadores, en un futuro se realizarán entrevistas con los clientes para definir de forma exacta toda la información relevante que se desee almacenar de los levantadores.

|  |  |
| --- | --- |
| **Config** | |
| **Descripción Entidad**: Esta entidad tiene una gran importancia dentro del sistema ya que se utiliza para representar las distintas configuraciones que sustentan el comportamiento del programa. De la entidad config y sus relaciones se sustenta la lógica del negocio del proyecto. | |
| **Atributo** | **Descripción Atributo** |
| version | Este atributo representa la versión de cada configuración registrada y es utilizado como identificador único dentro del contexto de la base de datos |

Tabla 4

**Relaciones:**

**Relación de 1-m** entre las entidades **Config** y **Herramienta**: Esta relación indica que una configuración tiene registrada muchas herramientas, y una herramienta pertenece a una única configuración.

**Relación de 1-m** entre las entidades **Config** y **Sistema**: Esta relación indica que una configuración tiene registrado muchos sistemas, y un sistema pertenece a una única configuración.

**Relación de 1-m** entre las entidades **Config** y **Indice\_Calculable**: Esta relación indica que una configuración tiene registrado muchos índices calculables, y un índice calculable pertenece a una única configuración.

|  |  |
| --- | --- |
| **Herramienta** | |
| **Descripción Entidad**: Esta entidad representa una abstracción de las distintas herramientas dentro de la metodología de evaluación técnica de una edificación. En esta entidad se definen aspectos generales y comunes de cada herramienta. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna a la herramienta en el momento en el que se registra |
| nombre | Representa el nombre por el que va a ser conocida la herrameinta, por ejemplo: (Herramienta de Análisis estructural con índices variables) |
| config\_version | Este atributo identifica la configuración a la cual pertenece la herramienta |

Tabla 5

|  |  |
| --- | --- |
| **Herramienta\_Analisis\_Criticidad** | |
| **Descripción Entidad**: Esta entidad representa la definición de un tipo de herramienta en concreto, en este caso una herramienta basada en la metodología utilizada en la herramienta de la confiabilidad operacional “Análisis de Criticidad” investigada en el trabajo de maestría de Quiñones. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna a la herramienta en el momento en el que se registra |
| nombre | Representa el nombre por el que va a ser conocida la herramienta, por ejemplo: (Herramienta de Análisis estructural con índices variables) |
| config\_version | Este atributo identifica la configuración a la cual pertenece la herramienta |

Tabla 6

**Relaciones:**

**Relación de 1-m** entre las entidades **Herramienta\_Analisis\_Criticidad** y **Campo**: Esta relación indica que una configuración tiene registrado muchos campos, y un campo pertenece a una única configuración.

|  |  |
| --- | --- |
| **Herramienta\_Metodo\_Impactos** | |
| **Descripción Entidad**: Esta entidad representa la definición de un tipo de herramienta en concreto, en este caso una herramienta basada en la metodología utilizada en la herramienta de la confiabilidad operacional “Método de los Impactos” investigada en el trabajo de maestría de Quiñones. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna a la herramienta en el momento en el que se registra |
| nombre | Representa el nombre por el que va a ser conocida la herramienta, por ejemplo: (Herramienta de Análisis estructural con índices variables) |
| config\_version | Este atributo identifica la configuración a la cual pertenece la herramienta |

Tabla 6

|  |  |
| --- | --- |
| **Herramienta\_Tecnica\_Paretos** | |
| **Descripción Entidad**: Esta entidad representa la definición de un tipo de herramienta en concreto, en este caso una herramienta basada en la metodología utilizada en la herramienta de la confiabilidad operacional “Técnica de Paretos” investigada en el trabajo de maestría de Quiñones. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna a la herramienta en el momento en el que se registra |
| nombre | Representa el nombre por el que va a ser conocida la herramienta, por ejemplo: (Herramienta de Análisis estructural con índices variables) |
| config\_version | Este atributo identifica la configuración a la cual pertenece la herramienta |

Tabla 8

|  |  |
| --- | --- |
| **Campo** | |
| **Descripción Entidad**: Esta entidad representa los campos definidos en la herramienta tipo Análisis Criticidad. Esto campos son elementos de una edificación a los cuales se les denota un nivel de importancia dentro del contexto de la evaluación del estado técnico de la edificación | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al campo en el momento en el que se registra |
| nombre | Representa el nombre del campo, por ejemplo: (Salud, Estructura, Estética) |
| nivel\_importancia | Representa el nivel de importancia del campo dentro de la evaluación de la edificación oscila dentro de estos valores (1: Baja, 2: Media, 3: Alta) |
| herramienta\_analisis\_criticidad\_id | Este atributo identifica la herramienta análisis criticidad a la cual pertenece el campo |

Tabla 9

|  |  |
| --- | --- |
| **Indice\_Calculable** | |
| **Descripción Entidad**: Esta entidad engloba los índices empleados en los cálculos inherentes al proceso de evaluación del estado técnico de una edificación, según la metodología de Quiñones. Mediante esta entidad, es posible definir dichos índices y ajustar sus características, lo que contribuye a hacer el programa más adaptable y versátil. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al índice calculable en el momento en el que se registra |
| nombre | Representa el nombre del índice cálculable, por ejemplo: (Índice Impactos, Índice Criticidad) |
| calculo\_id | Este atributo identifica al cálculo al cual hace referencia el índice calculable |
| config\_version | Este atributo identifica la configuración a la cual pertenece el índice calculable |

Tabla 10

En el contexto de los índices calculables, cada uno de ellos está acompañado por una serie de indicadores. Estos indicadores pueden estar definidos dentro de un intervalo o no, según el tipo específico del índice calculable en cuestión.

|  |  |
| --- | --- |
| **Indice\_Calculable\_Inervalo** | |
| **Descripción Entidad**: Esta entidad representa un tipo de índice calculable en concreto en este caso un índice calculable por intervalos, definido por indicadores. Donde cada indicador de dicho índice va a estar definido dentro de un intervalo. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al índice calculable en el momento en el que se registra |
| nombre | Representa el nombre del índice cálculable, por ejemplo: (Índice Impactos, Índice Criticidad) |
| calculo\_id | Este atributo identifica al cálculo al cual hace referencia el índice calculable |
| config\_version | Este atributo identifica la configuración a la cual pertenece el índice calculable |

Tabla 11

**Relaciones:**

**Relación de 1-m** entre las entidades **Indice\_Calculable\_Inervalo** y **Indicador\_Intervalo**: Esta relación indica que un índice calculable por intervalo tiene registrado muchos indicadores por intervalo, y un indicador por intervalo pertenece a un único índice calculable por intervalo.

|  |  |
| --- | --- |
| **Indice\_Calculable\_Sin\_Inervalo** | |
| **Descripción Entidad**: Esta entidad representa un tipo de índice calculable en concreto en este caso un índice calculable sin intervalos, definido por indicadores. Donde cada indicador de dicho índice va a estar definido por su propio valor ajeno a cualquier intervalo. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al índice calculable en el momento en el que se registra |
| nombre | Representa el nombre del índice cálculable, por ejemplo: (Índice Impactos, Índice Criticidad) |
| calculo\_id | Este atributo identifica al cálculo al cual hace referencia el índice calculable |
| config\_version | Este atributo identifica la configuración a la cual pertenece el índice calculable |

Tabla 12

**Relaciones:**

**Relación de 1-m** entre las entidades **Indice\_Calculable\_Sin\_Inervalo** y **Indicador\_Sin\_Intervalo**: Esta relación indica que un índice calculable sin intervalo tiene registrado muchos indicadores sin intervalo, y un indicador sin intervalo pertenece a un único índice calculable sin intervalo.

|  |  |
| --- | --- |
| **Indicador** | |
| **Descripción Entidad**: Esta entidad representa los valores asociados a cada índice calculable, la entidad indicador está conformada por el valor del indicador y por su significado. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al indicador en el momento en el que se registra |
| nombre | Representa el significado del indicador, por ejemplo: (Alta, Media, Baja) |
| valor | Representa el valor numérico del indicador en forma de índice por ejemplo: (1, 2, 3) |

Tabla 13

|  |  |
| --- | --- |
| **Indicador\_Intervalo** | |
| **Descripción Entidad**: Esta entidad representa un tipo de indicador en concreto en este caso un indicador por intervalos, el cual define el valor y su significado dentro de un intervalo. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al indicador en el momento en el que se registra |
| nombre | Representa el significado del indicador, por ejemplo: (Alta, Media, Baja) |
| valor | Representa el valor numérico del indicador en forma de índice, por ejemplo: (1, 2, 3) |
| limite\_inferior | Representa el extremo inferior del intervalo |
| limite\_superior | Representa el extremo superior del intervalo |
| indice\_calculable\_intervalo | Este atributo identifica al índice calculable por intervalo al que pertenece el indicador por intervalo |

Tabla 14

|  |  |
| --- | --- |
| **Indicador\_Sin\_Intervalo** | |
| **Descripción Entidad**: Esta entidad representa un tipo de indicador en concreto en este caso un indicador sin intervalos, el cual define el valor y su significado ajeno a cualquier intervalo. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al indicador en el momento en el que se registra |
| nombre | Representa el significado del indicador, por ejemplo: (Alta, Media, Baja) |
| valor | Representa el valor numérico del indicador en forma de índice, por ejemplo: (1, 2, 3) |
| indice\_calculable\_sin\_intervalo | Este atributo identifica al índice calculable sin intervalo al que pertenece el indicador sin intervalo |

Tabla 15

|  |  |
| --- | --- |
| **Sistema** | |
| **Descripción Entidad**: Esta entidad representa los distintos sistemas componentes de una Edificación. La definición de estos sistemas en la base de datos dentro del contexto configurable representa los sistemas que van a formar parte de la entidad levantamiento dentro del proceso de diagnóstico de la edificación. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al sistema en el momento en el que se registra |
| nombre | Representa el nombre por el que va a ser conocido el Sistema, por ejemplo: (Estructural, No Estructural, Hidrosanitario) |
| config\_version | Este atributo identifica la configuración a la cual pertenece el Sistema |
| herramienta\_id | Este atributo identifica la herramienta que fue seleccionada para ser usada en el Sistema |

Tabla 16

**Relaciones:**

**Relación de 1-m** entre las entidades **Sistema** y **Subsistema**: Esta relación indica que un sistema está compuesto por muchos subsistemas y un subsistema forma parte de un único sistema.

**Relación de m-1** entre las entidades **Sistema** y **Herramienta**: Esta relación indica que un sistema está asociado a una herramienta y una herramienta puede estar asociada a muchos sistemas.

|  |  |
| --- | --- |
| **Subsistema** | |
| **Descripción Entidad**: Esta entidad representa los distintos subsistemas componentes de un sistema de una Edificación. La definición de estos subsistemas en la base de datos dentro del contexto configurable representa los subsistemas que van a formar parte de la entidad sistema de un levantamiento dentro del proceso de diagnóstico de la edificación. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al subsistema en el momento en el que se registra |
| nombre | Representa el nombre por el que va a ser conocido el subsistema, por ejemplo: (Columna, Viga) |
| sistema\_id | Este atributo identifica el sistema al cual pertenece el subsistema |

Tabla 17

**Relaciones:**

**Relación de 1-m** entre las entidades **Subsistema** y **material**: Esta relación indica que un subsistema está compuesto por muchos materiales y un material forma parte de un único subsistema.

|  |  |
| --- | --- |
| **Material** | |
| **Descripción Entidad**: Esta entidad representa los distintos materiales componentes de un subsistema de un sistema de una Edificación. La definición de estos materiales en la base de datos dentro del contexto configurable representa los materiales que van a formar parte de la entidad subsistema de un sistema del levantamiento dentro del proceso de diagnóstico de la edificación. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al material en el momento en el que se registra |
| nombre | Representa el nombre por el que va a ser conocido el material, por ejemplo: (Piedra, Acero) |
| subsistema\_id | Este atributo identifica el subsistema al cual pertenece el material |

Tabla 18

**Relaciones:**

**Relación de 1-m** entre las entidades **Material** y **TipoDeterioro**: Esta relación indica que un material posee muchos tipos de deterioro asociados y un tipo de deterioro está asociado a un único material.

|  |  |
| --- | --- |
| **TipoDeterioro** | |
| **Descripción Entidad**: Esta entidad representa los distintos tipos de deterioro asociados al material de un subsistema de un sistema de una Edificación. La definición de estos tipos de deterioro en la base de datos dentro del contexto configurable representa los tipos de deterioro asociados a la entidad material de un subsistema de un sistema del levantamiento dentro del proceso de diagnóstico de la edificación. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al tipo de deterioro en el momento en el que se registra |
| nombre | Representa el nombre por el que va a ser conocido el tipo de deterioro, por ejemplo: (Grieta, Humedad) |
| material\_id | Este atributo identifica el material al cual pertenece el tipo de deterioro |

Tabla 19

**Relaciones:**

**Relación de 1-m** entre las entidades **TipoDeterioro** y **Causa**: Esta relación indica que un tipo de deterioro puede ser originado por varias causas y una causa puede originar un único tipo de deterioro.

**Relación de 1-m** entre las entidades **TipoDeterioro** y **CampoDefino**: Esta relación indica que un tipo de deterioro está caracterizado por varios campos definidos y un campo definido caracteriza a un único tipo de deterioro.

|  |  |
| --- | --- |
| **TipoDeterioroAnalisisCriticidad** | |
| **Descripción Entidad** Esta entidad representa la definición de un tipo de deterioro en concreto, en este caso un tipo de deterioro basado en los cálculos inherentes de la herramienta análisis criticidad. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al tipo de deterioro en el momento en el que se registra |
| nombre | Representa el nombre por el que va a ser conocido el tipo de deterioro, por ejemplo: (Grieta, Humedad) |
| material\_id | Este atributo identifica el material al cual pertenece el tipo de deterioro |

Tabla 20

**Relaciones:**

**Relación de m-m** entre las entidades **TipoDeterioroAnalisisCriticidad** y **Campo**: Esta relación indica que un tipo de deterioro análisis criticidad puede afectar a muchos campos y un campo puede ser afectado por muchos tipos de deterioro análisis criticidad.

|  |  |
| --- | --- |
| **TipoDeterioroMetodoImpactos** | |
| **Descripción Entidad** Esta entidad representa la definición de un tipo de deterioro en concreto, en este caso un tipo de deterioro basado en los cálculos inherentes de la herramienta método de los impactos. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al tipo de deterioro en el momento en el que se registra |
| nombre | Representa el nombre por el que va a ser conocido el tipo de deterioro, por ejemplo: (Grieta, Humedad) |
| material\_id | Este atributo identifica el material al cual pertenece el tipo de deterioro |

Tabla 21

|  |  |
| --- | --- |
| **TipoDeterioroTecnicaParetos** | |
| **Descripción Entidad** Esta entidad representa la definición de un tipo de deterioro en concreto, en este caso un tipo de deterioro basado en los cálculos inherentes de la herramienta técnica de paretos. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al tipo de deterioro en el momento en el que se registra |
| nombre | Representa el nombre por el que va a ser conocido el tipo de deterioro, por ejemplo: (Grieta, Humedad) |
| material\_id | Este atributo identifica el material al cual pertenece el tipo de deterioro |

Tabla 22

|  |  |
| --- | --- |
| **Causa** | |
| **Descripción Entidad** Esta entidad representa las causas de que definen el nacimiento de un tipo de deterioro. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna a la causa en el momento en el que se registra |
| nombre | Representa el nombre de la causa |
| tipo\_deterioro\_id | Este atributo identifica al tipo de deterioro que se encuentra asociado con la causa |

Tabla 23

|  |  |
| --- | --- |
| **Campo\_Definido** | |
| **Descripción Entidad** Esta entidad representa los campos definidos en cada tipo de deterioro. Esto campos representan el modo de almacenamiento de la información suministrada por el usuario del software. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al campo definido en el momento en el que se registra |
| nombre | Representa el nombre del campo definido, por ejemplo: (Ancho, Largo, Profundidad) |
| tipo | Representa el tipo de datos que va almacenar el campo definido, por ejemplo: (Numero, Texto, Fecha) |
| tipo\_deterioro\_id | Este atributo identifica al tipo de deterioro que se encuentra asociado con el campo definido |

Tabla 24

|  |  |
| --- | --- |
| **Deterioro** | |
| **Descripción Entidad** Esta entidad representa los deterioros identificados en el proceso de evaluación del estado técnico de la edificación, estos deterioros son identificados en componentes integrantes de una edificación. | |
| **Atributo** | **Descripción Atributo** |
| id | Un identificador único que se le asigna al deterioro en el momento en el que se registra |
| tipo\_deterioro\_id | Este atributo identifica al tipo de deterioro al cual está asociado el deterioro |
| material\_id | Este atributo identifica al material donde fue identificado el deterioro |
| Subsistema\_id | Este atributo identifica al subsistema donde fue identificado el deterioro |
| Sistema\_id | Este atributo identifica al sistema donde fue identificado el deterioro |
| levantamiento\_id | Este atributo identifica al levantamiento al cual pertenece el deterioro |

Tabla 25

**Relaciones:**

**Relación de m-1** entre las entidades **Deterioro** y **TipoDeterioro**: Esta relación indica que un deterioro está asociado a un único tipo de deterioro y un tipo de deterioro tiene asociado muchos deterioros.

**Relación de m-1** entre las entidades **Deterioro** y **Material**: Esta relación indica que un deterioro fue identificado en un único material y un material puede tener identificados muchos deterioros.

**Relación de m-1** entre las entidades **Deterioro** y **Subsistema**: Esta relación indica que un deterioro fue identificado en un único subsistema y un subsistema puede tener identificados muchos deterioros.

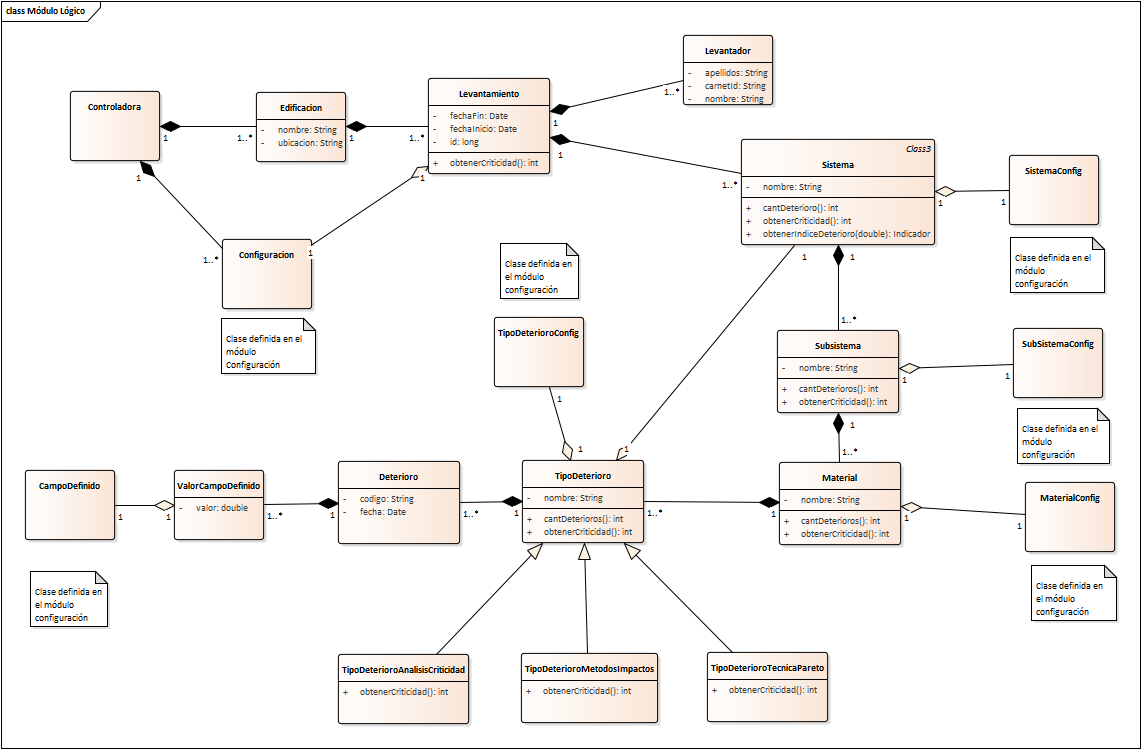
**Relación de m-1** entre las entidades **Deterioro** y **Sistema**: Esta relación indica que un deterioro fue identificado en un único sistema y un sistema puede tener identificados muchos deterioros.

**Relación de m-m** entre las entidades **Deterioro** y **CampoDefinido**: Esta relación indica que un deterioro está caracterizado por varios campos definidos con un valor y un campo definido puede caracterizar a varios deterioros con un valor.

**Modelo de Dominio**

En el contexto del desarrollo de software, el modelo de dominio se erige como un pilar fundamental para comprender y estructurar los datos que conforman un sistema. Su propósito es claro: dar sentido a la información almacenada y definir un mapa conceptual para el procesamiento de los datos. A continuación, se presentará la propuesta de modelo de dominio del proyecto.

El modelo de dominio de este proyecto tiene como objetivo organizar la información para facilitar su procesamiento y la realización de los cálculos. Para ello una vez cargada la información de la base de datos se estructurará de la siguiente forma:



**Estructura de un Levantamiento:**

En el contexto del modelo de datos, un Levantamiento se considera una entidad vinculada a una configuración. Esta configuración constituye el núcleo del levantamiento y define su estructura, proporcionando el marco para las operaciones que se llevan a cabo en él. En el modelo de dominio, se emplea la información registrada en la configuración, así como los datos sobre los deterioros específicos del levantamiento. A partir de estos elementos, se crea una estructura jerárquica en forma de árbol, que se desglosa de la siguiente manera:

**Sistemas:** Cada levantamiento se compone de varios sistemas.

**Subsistemas:** Dentro de cada sistema, existen subsistemas adicionales.

**Materiales:** A su vez, cada subsistema está asociado con diferentes materiales.

**Tipos de Deterioro Asociados:** Para cada material, se identifican los tipos de deterioro relacionados.

**Lista de Deterioros Identificados:** Finalmente, se registra una lista de los deterioros específicos encontrados en el levantamiento, clasificados según su tipo.

Esta estructura permite realizar los cálculos para determinar la criticidad de cada componente dentro del levantamiento. Así, se le brinda al front end la información organizada para evaluar la importancia relativa de cada parte en el contexto general de la estructura.

Para la realización de los cálculos:

**Índice de Criticidad Tipo Deterioro:**

El cálculo del índice de criticidad se implementa en el método obtenerIndiceCriticidad() de la clase Tipo de Deterioro, este método tendrá una implementación diferente de acuerdo al Tipo de Deterioro en cuestión:

**Cálculo Índice criticidad para Tipo de Deterioro de Análisis de Criticidad**

Obtener cada uno de los índices correspondientes que intervienen en el cálculo:

Índice Impacto (ver 2.2)

Índice de detectabilidad (ver 2.3)

Índice frecuencia (ver 2.4)

Luego el índice de criticidad será la multiplicación de estos índices:

Índice de criticidad = Índice Impacto \* Índice de detectabilidad \* Índice frecuencia

**Cálculo Índice Impacto Tipo de Deterioro de Análisis de Criticidad:**

El cálculo del índice de impacto se lleva a cabo mediante el método obtenerIndiceImpacto() del tipo de deterioro de análisis de criticidad. Para realizar este cálculo, se consideran los campos afectados por el tipo de deterioro registrados en la configuración. El proceso consiste en sumar la importancia de estos campos y, posteriormente, determinar el índice de impacto evaluando esta suma dentro de los intervalos definidos en el índice calculable registrado en la configuración para dicho cálculo.

**Cálculo Índice Frecuencia Tipo de Deterioro de Análisis de Criticidad:**

El cálculo del índice de frecuencia se realiza mediante el método obtenerIndiceFrecuencia() del tipo de deterioro de análisis de criticidad. Para llevar a cabo este cálculo, primero se determina la **cantidad total de deterioros** del mismo tipo que pertenecen al sistema al que corresponde dicho tipo de deterioro. El objetivo es calcular el **porcentaje de ocurrencia** de este tipo de deterioro en el mencionado sistema. Luego, para determinar el índice de frecuencia, evaluamos ese porcentaje dentro de los intervalos definidos en el índice calculable registrado en la configuración para este cálculo. Para poder realizar este cálculo fue necesario establecer una relación bidireccional entre la clase Tipo de Deterioro y Sistema.

**Cálculo Índice Detectabilidad Tipo de Deterioro de Análisis de Criticidad:**

El cálculo del índice de detectabilidad se realiza mediante el método obtenerIndiceImpacto() del tipo de deterioro en el análisis de criticidad. El procedimiento consiste en evaluar el valor de detectabilidad previamente configurado para ese tipo de deterioro y evaluarlo en el índice calculado sin inérvalo registrado específicamente para este cálculo.

Todos estos cálculos se basan en la configuración seleccionada para el levantamiento. Por lo tanto, es fundamental establecer una relación bidireccional entre el tipo de deterioro y el levantamiento al que pertenece, dado que el levantamiento está asociado a la configuración.

**Patrones de Diseño utilizados**

**Patrón Modelo-Vista-Controlador (MVC)**

Para el desarrollo del proyecto se decidió usar el patrón **Modelo-Vista-Controlador (MVC)** un enfoque arquitectónico ampliamente utilizado en el desarrollo de aplicaciones. Popularizado por Martin Fowler y se describe en detalle en el libro "Patterns of Enterprise Application Architecture. Su objetivo principal es separar las responsabilidades dentro de una aplicación, lo que facilita la modularidad de la solución, la reutilización de código y el mantenimiento. A continuación. Se divide en 3 componentes principales [12]:

**Modelo (Model):**

El Modelo representa la lógica de negocio y la gestión de datos.

Contiene la representación interna de los datos y las operaciones relacionadas con ellos.

Puede incluir clases para acceder a la base de datos, realizar cálculos, aplicar reglas de negocio y validar datos.

El Modelo no sabe nada sobre la interfaz de usuario ni cómo se presentan los datos.

**Vista (View):**

La Vista es la parte visible de la aplicación.

Representa la interfaz gráfica de usuario (GUI).

Su función principal es mostrar datos al usuario y recibir interacciones.

No contiene lógica de negocio; simplemente muestra lo que el Controlador le proporciona.

**Controlador (Controller):**

El Controlador actúa como intermediario entre la Vista y el Modelo.

Recibe las solicitudes del usuario desde la Vista (por ejemplo, un clic en un botón).

Interactúa con el Modelo para obtener o actualizar datos.

Decide qué Vista mostrar al usuario en función de la acción solicitada.

Puede realizar transformaciones en los datos antes de enviarlos a la Vista.

A continuación, se describe el flujo de trabajo del patrón [12]:

1. El usuario interactúa con la Vista (por ejemplo, hace clic en un botón).
2. El Controlador recibe la solicitud y decide qué acción tomar.
3. El Controlador se comunica con el Modelo para obtener o actualizar datos.
4. El Modelo procesa la solicitud y devuelve los datos al Controlador.
5. El Controlador selecciona la Vista apropiada y le envía los datos.
6. La Vista muestra los datos al usuario.

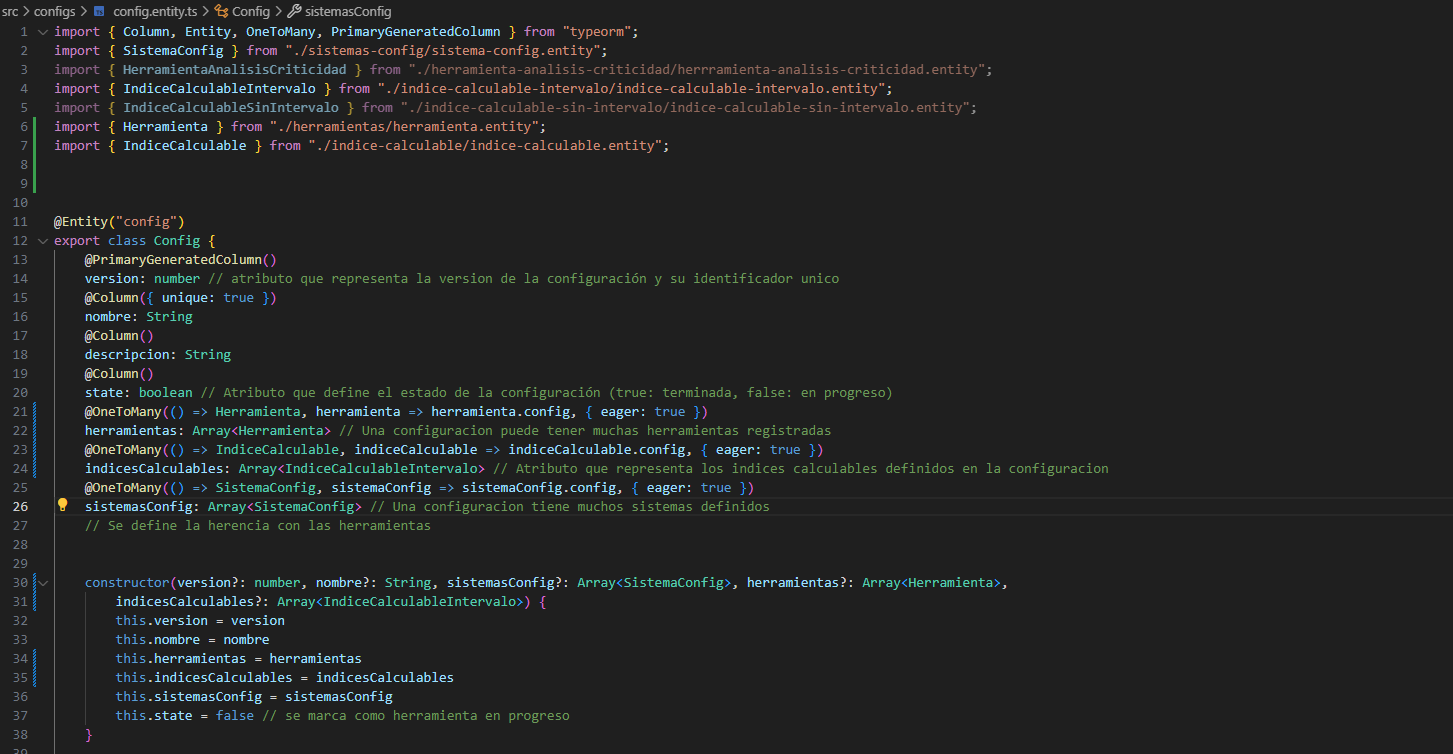
Algunas de las ventajas de usar este patrón son:

* Separación de preocupaciones: Cada componente tiene una responsabilidad clara.
* Reutilización de código: Los componentes son independientes y pueden reutilizarse en diferentes partes de la aplicación.
* Facilita el mantenimiento: Cambios en la lógica de negocio no afectan directamente la interfaz de usuario.

**¿Como se aplicó el patrón en el proyecto?**

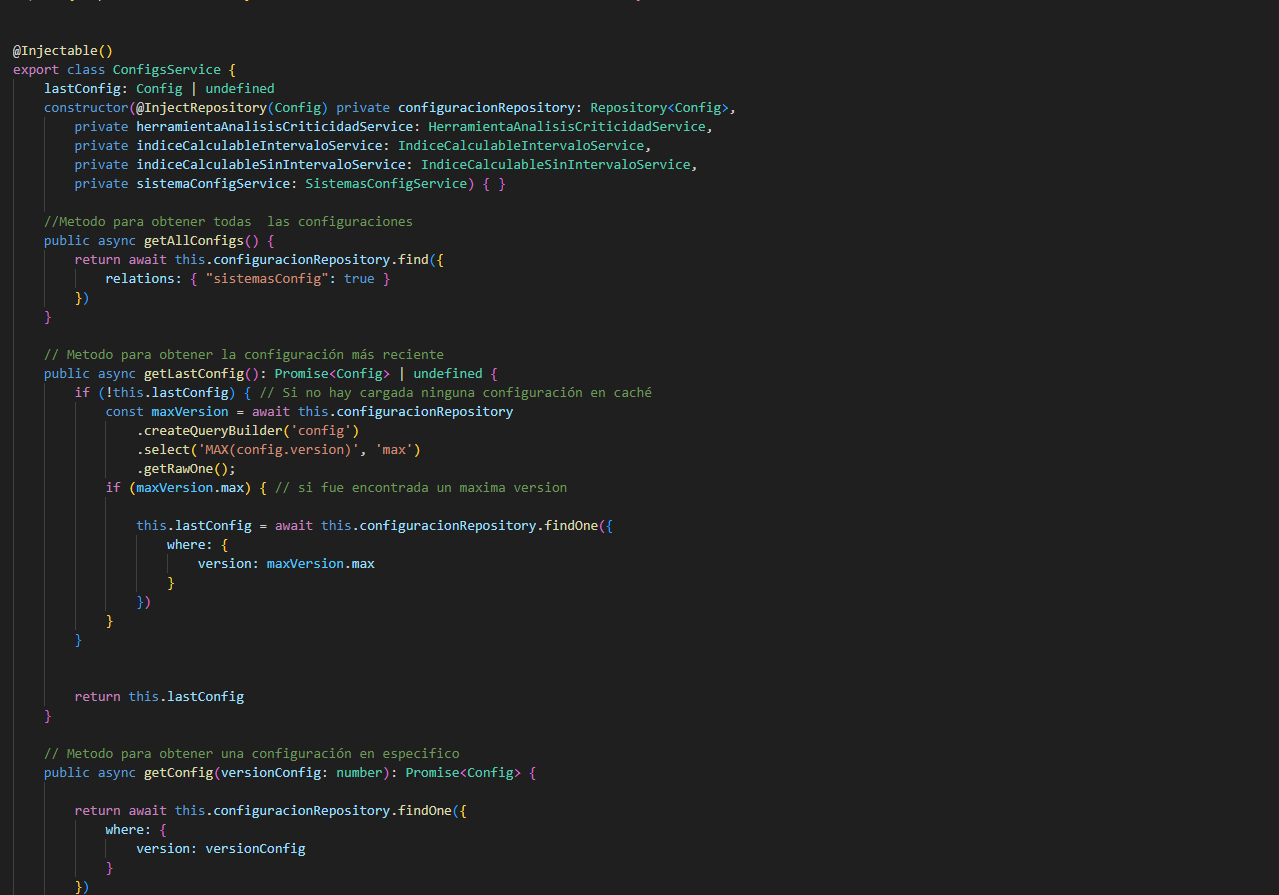
**El modelo** está representado por el backend de la aplicación, dentro de este más específicamente por las clases **Entity** responsables de definir la estructura y las relaciones entre los datos y las clases **Service** que representan la lógica de negocio y se encargan de interactuar con los datos.

Aquí un ejemplo de como se implementaron algunas de estas clases:



ConfigEntity

La clase Entity Config define la estructura y los datos que van hacer almacenados de una configuración.

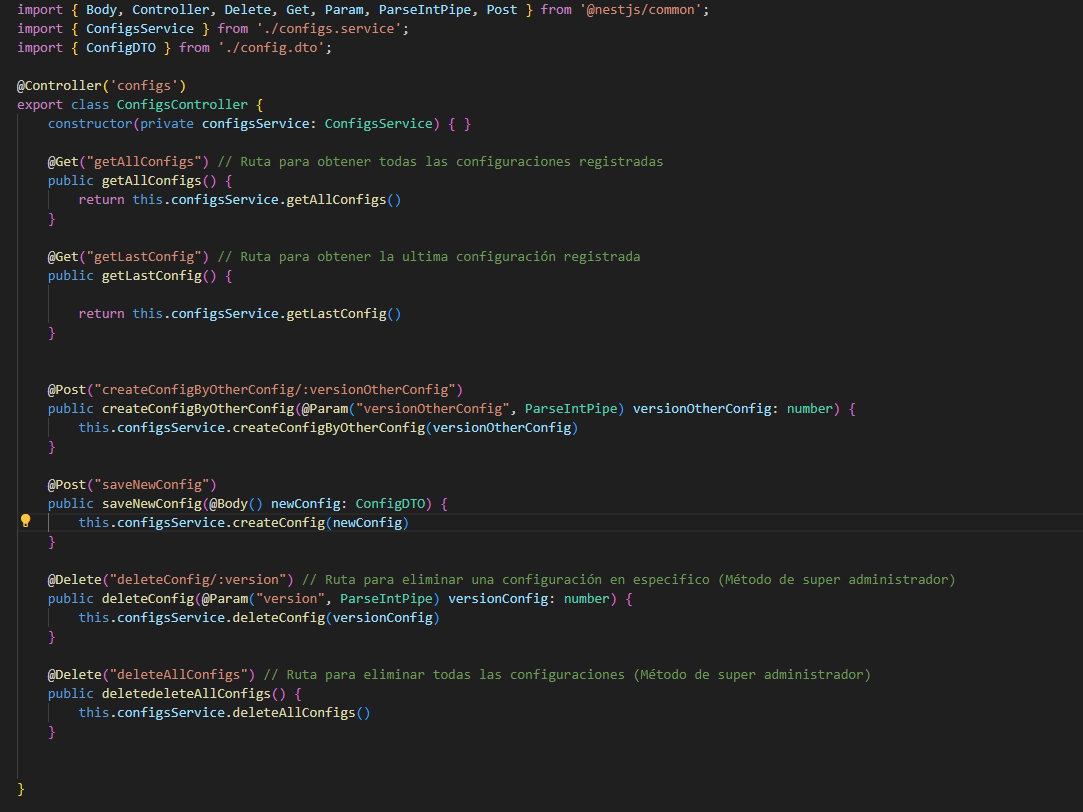


ConfigService

La clase ConfigService provee operaciones relacionadas con los datos defendidos en la entidad de la configuración como crear, leer, actualizar y eliminar entidades de configuración.

**El controlador** se representa mediante las clases Controller en el backend de la aplicación. Estas clases son responsables de gestionar las solicitudes entrantes y proporcionar respuestas al cliente. Para lograrlo, los controladores utilizan instancias de los servicios, a los cuales consultan para realizar las operaciones solicitadas por la vista. En resumen, las clases Controller actúan como el cerebro del proyecto.

A continuación, se presenta un ejemplo de una clase Controller:



ConfigController

La clase ConfigController define las rutas de solicitud mediante métodos HTTPS. En este contexto, cada método representa una operación solicitada por la vista, que se ejecutará realizando una consulta al servicio correspondiente, en este caso, el servicio ConfigService.

**La vista** representada por el front end de la aplicación en este caso el cliente web y móvil, ambos se encargan de presentar el contenido a través de una interfaz de usuario, realizando distintas peticiones HTTPS a los controladores de ruta del backend tanto para obtener y mostrar datos como para insertarlos, modificarlos y eliminarlos. El cliente web hace uso de graficas para la visualización de los datos de los levantamientos recibidos del backend, así como también de formularios diseñados para la gestión de las configuraciones del sistema, mientras que la aplicación móvil centra su labor en la recopilación de datos establecidos por la configuración recibida del backend.

**Patrón Inyección de Dependencias**

**La inyección de dependencias** (Dependency Injection, DI) es un patrón de diseño orientado a objetos que se utiliza para desacoplar componentes y promover la reutilización del código. En lugar de que los objetos creen sus propias dependencias, estas se proporcionan externamente, lo que facilita la extensibilidad y modificación de la aplicación sin alterar su código base [12].

**Concepto** [12]**:**

La inyección de dependencias es un patrón que busca desacoplar componentes y promover la reutilización del código.

En lugar de que los objetos creen sus propias dependencias, estas se proporcionan externamente.

El objetivo es facilitar la extensibilidad y modificación de la aplicación sin alterar su código base.

**Implementación** [12]**:**

En la inyección de dependencias, los objetos no crean directamente las instancias de las clases que necesitan.

En cambio, se suministran objetos a una clase desde una fuente externa (un “contenedor DI” o “contenedor IoC”).

Estos objetos cumplen contratos que nuestras clases requieren para funcionar correctamente.

**Beneficios** [12]**:**

Desacopla componentes: Los objetos no necesitan conocer los detalles de cómo se crean sus dependencias.

Facilita las pruebas unitarias: Se pueden proporcionar implementaciones simuladas o de prueba para las dependencias.

Mejora el modularidad y la flexibilidad del diseño.

**¿Como se aplicó el patrón en el proyecto?**

Dentro del contexto del backend de la aplicación el patrón de inyección de dependencias es utilizado para varios procedimientos:

1. Proveer a los controladores de lo servicios necesarios para satisfacer las solicitudes de la vista para realizar eso Nest.js con el decorador @Iyectable() declara a las clases Service que pueden ser utilizadas por el contenedor IoC de Nest.js, luego en los controladores, se pueden declarar dependencias en el token mediante la inyección en el constructor, el resultado de esto es que el contenedor IoC de Nest.js instancia un controlador, busca sus dependencias, al encontrar esas dependencia, realiza una búsqueda en el token de los Service y devuelve las clase Service inyectadas a ese controlador, si está configurado como ámbito singleton (el comportamiento predeterminado), Nest.js crea una instancia de Service o devuelve una instancia existente si ya está en caché. En la figura 2.3, se el controlador Config contienen una instancia del Serivcio Config anotado como @Iyenctable ver figura 2.2.
2. Trabajo con TypeORM para acceder a repositorios de manera eficiente. Los repositorios son componentes que proporcionan funcionalidades típicas de acceso a datos y operaciones de escritura en las tablas de la base de datos. Para inyectar un repositorio en un servicio, utiliza el decorador @InjectRepository(), este decorador instancia automáticamente el repositorio de la entidad especificada cada vez que vez el servicio es instanciado en la figura 2.22 se puede apreciar la inyección de un repositorio de la entidad Config.