**Universidad de Cartagena**

**Facultad de Ingeniería de la Universidad de Cartagena**



**ACTIVIDAD I - UNIDAD I**

**PROGRAMACIÓN ORIENTADA A OBJETOS**

**Proyecto**

**Procesos Políticos**

**EMMANUEL GUTIERREZ GARCIA**

**Código 7502320013**

**Profesor**

**John Carlos Arrieta Arrieta**

**Cartagena – Colombia**

**Marzo de 2024**

Contenido

[INTRODUCCIÓN 5](#_Toc162902436)

[OBJETIVOS 5](#_Toc162902437)

[JUSTIFICACIÓN 5](#_Toc162902438)

[LISTA DE REQUERIMIENTOS FUNCIONALES 6](#_Toc162902439)

[CLASES QUE COMPONEN EL SISTEMA DE ELECCION ELECTORAL 7](#_Toc162902440)

[RELACIONES ENTRE CLASES 35](#_Toc162902441)

[MODELAR EN UML EL RESPECTIVO DIAGRAMA DE CLASE 36](#_Toc162902442)

[DEL DIAGRAMA AL CÓDIGO UML 37](#_Toc162902444)

[ENLACE DE GITHUB 41](#_Toc162902445)

[BIBLIOGRAFÍA 41](#_Toc162902446)

# INTRODUCCIÓN

En el proceso del desarrollo de sistemas empresariales, la implementación efectiva de un modelo orientado a objetos se convierte en un elemento fundamental para la gestión integral de información. En este sentido, el presente trabajo aborda la problemática planteada por un grupo empresarial que busca establecer un sistema que englobe tanto a las empresas que conforman su holding como a sus vendedores y asesores. Este enfoque requiere una estructura de datos bien definida que permita gestionar de manera eficiente la jerarquía piramidal de vendedores, la diversidad de sectores de mercado en los que operan las empresas, la distribución geográfica de las operaciones, así como el apoyo multidisciplinario proporcionado por los asesores. Por consiguiente, el diseño y la implementación de un sistema basado en programación orientada a objetos se erigen como pilares fundamentales para la consecución de los objetivos empresariales y la optimización de procesos internos.

# OBJETIVOS

* Consolidar los conocimientos vistos en los materiales.
* Mejorar el proceso de diseño de un programa con el paradigma orientado a objetos.
* Comprender de manera completa el documento de los casos de uso e historias de usuario. Para poder implementar de manera correcta y apropiada la solución.
* Medir la eficiencia y eficacia del programa para poder plantear mejoras en su diseño, código o lógica.
* Diseñar la interfaz de los requerimientos con el fin de que seas de fácil uso para el usuario final

# JUSTIFICACIÓN

Es de vital importancia para las empresas del holding tener el control y total conocimiento sobre todos los reclutamientos asignaciones de asesores e información de todas las empresas del holding por lo cual el software se diseño para cumplir con las necesidades.

# LISTA DE REQUERIMIENTOS FUNCIONALES

**Creación de Perfiles de Vendedores:**

- Permitir la creación de perfiles de vendedores con campos para código, nombre y dirección.

**Jerarquía Piramidal de Vendedores:**

- Establecer una jerarquía piramidal donde cada vendedor pueda reclutar a otros vendedores.

- Registrar la fecha en que un vendedor recluta a otro.

**Restricción de Trabajo y Reclutamiento de Vendedores:**

- Limitar a cada vendedor a trabajar solo para una empresa.

- Restringir la capacidad de reclutamiento de vendedores a la empresa en la que el vendedor trabaja.

**Creación de Perfiles de Empresas:**

- Facilitar la creación de perfiles de empresas con campos para nombre, fecha de entrada al holding, facturación anual y número de vendedores.

- Permitir la asignación de áreas de mercado a cada empresa.

- Registrar la sede principal de cada empresa, incluyendo el país y la ciudad.

- Soportar la posibilidad de que una empresa tenga operaciones en múltiples países.

- Permitir que una empresa tenga su sede en un país distinto al de sus operaciones.

**Creación de Perfiles de Países:**

- Permitir la creación de perfiles de países con campos para nombre, PIB, población y capital.

**Creación de Perfiles de Asesores:**

- Posibilitar la creación de perfiles de asesores con campos para código, nombre, dirección y titulación.

- Permitir que un asesor cubra múltiples áreas de actividad del holding.

- Permitir que un área de actividad sea cubierta por varios asesores.

- Permitir que un asesor asesore a múltiples empresas.

- Registrar la fecha en que un asesor comienza a trabajar para una empresa en una determinada área.

**Sistema de Autenticación y Control de Acceso:**

- Implementar un sistema de autenticación para garantizar que solo usuarios autorizados puedan acceder al Sistema.

- Establecer diferentes niveles de acceso según el rol del usuario (por ejemplo, administrador, vendedor, asesor).

- Permitir la asignación y revocación de permisos de acceso a los datos según las funciones y responsabilidades de cada usuario.

**Gestión de Relaciones entre Vendedores y Empresas:**

- Registrar la relación laboral entre los vendedores y las empresas para garantizar que cada vendedor esté asociado únicamente a una empresa.

- Mantener un registro actualizado de los vendedores activos en cada empresa, así como de los vendedores que han sido reclutados por otros vendedores.

**Herramientas de Análisis para Evaluación de Rendimiento:**

- Proporcionar herramientas de análisis para evaluar el rendimiento de los vendedores, como el número de ventas realizadas o el volumen de facturación generado.

- Generar informes periódicos sobre la facturación anual y el número de vendedores de cada empresa para evaluar su desempeño dentro del holding.

**Gestión de Contratos y Asignaciones de Asesores:**

- Registrar los contratos entre los asesores y las empresas para especificar las responsabilidades y áreas de trabajo de cada asesor.

- Facilitar la asignación de asesores a empresas y áreas específicas según las necesidades del holding y las competencias de los asesores disponibles.

# CLASES QUE COMPONEN EL SISTEMA DE ELECCION ELECTORAL

**NOMBRE DE LA CLASE:**

● Usuario

**DESCRIPCIÓN:**

● Esta clase representa un usuario con sus respectivas propiedades como código, nombre, dirección, nombre de usuario, clave y perfil.

**PROPIEDADES DE LA CLASE:**

● codigo. Tipo: String. Es requerido: Sí.

● nombre. Tipo: String. Es requerido: Sí.

● direccion. Tipo: String. Es requerido: No.

● nombreUsuario. Tipo: String. Es requerido: No.

● clave. Tipo: String. Es requerido: No.

● perfil. Tipo: String. Es requerido: No.

**OPERACIONES (MÉTODOS) DE LA CLASE:**

● Constructor

- Descripción del método: Inicializa un nuevo objeto Usuario.

- Objetivo: Inicializar las propiedades del objeto.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Ninguno.

public Usuario()

● getCodigo

- Descripción del método: Devuelve el código del usuario.

- Objetivo: Obtener el código del usuario.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getCodigo()

● setCodigo

- Descripción del método: Establece el código del usuario.

- Objetivo: Asignar un valor al código del usuario.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String codigo.

public void setCodigo(String codigo)

● getNombre

- Descripción del método: Devuelve el nombre del usuario.

- Objetivo: Obtener el nombre del usuario.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getNombre()

● setNombre

- Descripción del método: Establece el nombre del usuario.

- Objetivo: Asignar un valor al nombre del usuario.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String nombre.

public void setNombre(String nombre)

● getDireccion

- Descripción del método: Devuelve la dirección del usuario.

- Objetivo: Obtener la dirección del usuario.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getDireccion()

● setDireccion

- Descripción del método: Establece la dirección del usuario.

- Objetivo: Asignar un valor a la dirección del usuario.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String direccion.

public void setDireccion(String direccion)

● getNombreUsuario

- Descripción del método: Devuelve el nombre de usuario del usuario.

- Objetivo: Obtener el nombre de usuario del usuario.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getNombreUsuario()

● setNombreUsuario

- Descripción del método: Establece el nombre de usuario del usuario.

- Objetivo: Asignar un valor al nombre de usuario del usuario.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String nombreUsuario.

public void setNombreUsuario(String nombreUsuario)

● getClave

- Descripción del método: Devuelve la clave del usuario.

- Objetivo: Obtener la clave del usuario.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getClave()

● setClave

- Descripción del método: Establece la clave del usuario.

- Objetivo: Asignar un valor a la clave del usuario.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String clave.

public void setClave(String clave)

● getPerfil

- Descripción del método: Devuelve el perfil del usuario.

- Objetivo: Obtener el perfil del usuario.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getPerfil()

● setPerfil

- Descripción del método: Establece el perfil del usuario.

- Objetivo: Asignar un valor al perfil del usuario.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String perfil.

public void setPerfil(String perfil)

**NOMBRE DE LA CLASE:**

● Vendedor

**DESCRIPCIÓN:**

● Esta clase representa un vendedor con sus respectivas propiedades como id, fecha de reclutamiento, id del líder y la empresa a la que pertenece.

**PROPIEDADES DE LA CLASE:**

● id. Tipo: Int. Es requerido: Sí.

● fechaReclutamiento. Tipo: Date. Es requerido: Sí.

● id\_lider. Tipo: Int. Es requerido: Sí.

● empresa. Tipo: Empresa. Es requerido: Sí.

**OPERACIONES (MÉTODOS) DE LA CLASE:**

● getId

- Descripción del método: Devuelve el ID del vendedor.

- Objetivo: Obtener el ID del vendedor.

- Ámbito: Público.

- Tipo de retorno: Int.

- Parámetros: Ninguno.

public Int getId()

● getFechaReclutamiento

- Descripción del método: Devuelve la fecha de reclutamiento del vendedor.

- Objetivo: Obtener la fecha de reclutamiento del vendedor.

- Ámbito: Público.

- Tipo de retorno: Date.

- Parámetros: Ninguno.

public Date getFechaReclutamiento()

● setFechaReclutamiento

- Descripción del método: Establece la fecha de reclutamiento del vendedor.

- Objetivo: Asignar un valor a la fecha de reclutamiento del vendedor.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Date fechaReclutamiento.

public void setFechaReclutamiento(Date fechaReclutamiento)

● getId\_lider

- Descripción del método: Devuelve el ID del líder del vendedor.

- Objetivo: Obtener el ID del líder del vendedor.

- Ámbito: Público.

- Tipo de retorno: Int.

- Parámetros: Ninguno.

public Int getId\_lider()

● setItem

- Descripción del método: Establece el ID del líder del vendedor.

- Objetivo: Asignar un valor al ID del líder del vendedor.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Int id\_lider.

public void setId\_lider(Int id\_lider)

● getEmpresa

- Descripción del método: Devuelve la empresa del vendedor.

- Objetivo: Obtener la empresa del vendedor.

- Ámbito: Público.

- Tipo de retorno: Empresa.

- Parámetros: Ninguno.

public Empresa getEmpresa()

● setEmpresa

- Descripción del método: Establece la empresa del vendedor.

- Objetivo: Asignar un valor a la empresa del vendedor.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Empresa empresa.

public void setEmpresa(Empresa empresa)

**NOMBRE DE LA CLASE:**

● Asesor

**DESCRIPCIÓN:**

● Esta clase representa un asesor con sus respectivas propiedades como id, titulación y fecha de inicio de trabajo.

**PROPIEDADES DE LA CLASE:**

● id. Tipo: Int. Es requerido: Sí.

● titulacion. Tipo: String. Es requerido: Sí.

● fechaInicioTrabajo. Tipo: Date. Es requerido: Sí.

**OPERACIONES (MÉTODOS) DE LA CLASE:**

● getId

- Descripción del método: Devuelve el ID del asesor.

- Objetivo: Obtener el ID del asesor.

- Ámbito: Público.

- Tipo de retorno: Int.

- Parámetros: Ninguno.

public Int getId()

● getTitulacion

- Descripción del método: Devuelve la titulación del asesor.

- Objetivo: Obtener la titulación del asesor.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getTitulacion()

● setTitulacion

- Descripción del método: Establece la titulación del asesor.

- Objetivo: Asignar un valor a la titulación del asesor.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String titulacion.

public void setTitulacion(String titulacion)

● getFechaTrabajo

- Descripción del método: Devuelve la fecha de inicio de trabajo del asesor.

- Objetivo: Obtener la fecha de inicio de trabajo del asesor.

- Ámbito: Público.

- Tipo de retorno: Date.

- Parámetros: Ninguno.

public Date getFechaTrabajo()

● setFechaTrabajo

- Descripción del método: Establece la fecha de inicio de trabajo del asesor.

- Objetivo: Asignar un valor a la fecha de inicio de trabajo del asesor.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Date fechaInicioTrabajo.

public void setFechaTrabajo(Date fechaInicioTrabajo)

**NOMBRE DE LA CLASE:**

● Empresa

**DESCRIPCIÓN:**

● Esta clase representa una empresa con sus respectivas propiedades como id, nombre, fecha de incorporación, facturación anual, número de vendedores, sede y una lista de vendedores asociados.

**PROPIEDADES DE LA CLASE:**

● id. Tipo: Int. Es requerido: Sí.

● nombre. Tipo: String. Es requerido: Sí.

● fechaIncorporacion. Tipo: Date. Es requerido: Sí.

● facturacionAnual. Tipo: Double. Es requerido: Sí.

● numeroVendedores. Tipo: Int. Es requerido: Sí.

● sede. Tipo: Sede. Es requerido: Sí.

● vendedores. Tipo: List<Vendedor>. Es requerido: No.

**OPERACIONES (MÉTODOS) DE LA CLASE:**

● getId

- Descripción del método: Devuelve el ID de la empresa.

- Objetivo: Obtener el ID de la empresa.

- Ámbito: Público.

- Tipo de retorno: Int.

- Parámetros: Ninguno.

public Int getId()

● getNombre

- Descripción del método: Devuelve el nombre de la empresa.

- Objetivo: Obtener el nombre de la empresa.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getNombre()

● setFechaIncorporacion

- Descripción del método: Establece la fecha de incorporación de la empresa.

- Objetivo: Asignar un valor a la fecha de incorporación de la empresa.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Date fechaIncorporacion.

public void setFechaIncorporacion(Date fechaIncorporacion)

● getFacturacionAnual

- Descripción del método: Devuelve la facturación anual de la empresa.

- Objetivo: Obtener la facturación anual de la empresa.

- Ámbito: Público.

- Tipo de retorno: Double.

- Parámetros: Ninguno.

public Double getFacturacionAnual()

● setFacturacionAnual

- Descripción del método: Establece la facturación anual de la empresa.

- Objetivo: Asignar un valor a la facturación anual de la empresa.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Double facturacionAnual.

public void setFacturacionAnual(Double facturacionAnual)

● getNumeroVendedores

- Descripción del método: Devuelve el número de vendedores de la empresa.

- Objetivo: Obtener el número de vendedores de la empresa.

- Ámbito: Público.

- Tipo de retorno: Int.

- Parámetros: Ninguno.

public Int getNumeroVendedores()

● setNumeroVendedores

- Descripción del método: Establece el número de vendedores de la empresa.

- Objetivo: Asignar un valor al número de vendedores de la empresa.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Int numeroVendedores.

public void setNumeroVendedores(Int numeroVendedores)

● getSede

- Descripción del método: Devuelve la sede de la empresa.

- Objetivo: Obtener la sede de la empresa.

- Ámbito: Público.

- Tipo de retorno: Sede.

- Parámetros: Ninguno.

public Sede getSede()

● getVendedores

- Descripción del método: Devuelve la lista de vendedores de la empresa.

- Objetivo: Obtener la lista de vendedores de la empresa.

- Ámbito: Público.

- Tipo de retorno: List<Vendedor>.

- Parámetros: Ninguno.

public List<Vendedor> getVendedores()

● toString

- Descripción del método: Devuelve una representación en formato String de la empresa.

- Objetivo: Obtener una representación en formato String de la empresa.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String toString()

**NOMBRE DE LA CLASE:**

● AsesoresSectores

DESCRIPCIÓN:

● Esta clase representa la relación entre un asesor y un sector del mercado.

**PROPIEDADES DE LA CLASE:**

● id. Tipo: Int. Es requerido: Sí.

● asesor. Tipo: Asesor. Es requerido: Sí.

● sector. Tipo: SectorDelMercado. Es requerido: Sí.

**OPERACIONES (MÉTODOS) DE LA CLASE:**

● getAsesor

- Descripción del método: Devuelve el asesor asociado.

- Objetivo: Obtener el asesor asociado.

- Ámbito: Público.

- Tipo de retorno: Asesor.

- Parámetros: Ninguno.

public Asesor getAsesor()

● setAsesor

- Descripción del método: Establece el asesor asociado.

- Objetivo: Asignar un valor al asesor asociado.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Asesor asesor.

public void setAsesor(Asesor asesor)

● getSector

- Descripción del método: Devuelve el sector del mercado asociado.

- Objetivo: Obtener el sector del mercado asociado.

- Ámbito: Público.

- Tipo de retorno: SectorDelMercado.

- Parámetros: Ninguno.

public SectorDelMercado getSector()

● setSector

- Descripción del método: Establece el sector del mercado asociado.

- Objetivo: Asignar un valor al sector del mercado asociado.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: SectorDelMercado sector.

public void setSector(SectorDelMercado sector)

**NOMBRE DE LA CLASE:**

● HistoricoAsesor

**DESCRIPCIÓN:**

● Esta clase representa el historial de un asesor en una empresa y un sector del mercado específicos.

**PROPIEDADES DE LA CLASE:**

● id. Tipo: Int. Es requerido: Sí.

● empresa. Tipo: Empresa. Es requerido: Sí.

● asesor. Tipo: Asesor. Es requerido: Sí.

● sector. Tipo: SectorDelMercado. Es requerido: Sí.

● fechaIncorporacion. Tipo: Date. Es requerido: Sí.

**OPERACIONES (MÉTODOS) DE LA CLASE:**

● getAsesor

- Descripción del método: Devuelve el asesor asociado al historial.

- Objetivo: Obtener el asesor asociado al historial.

- Ámbito: Público.

- Tipo de retorno: Asesor.

- Parámetros: Ninguno.

public Asesor getAsesor()

● setAsesor

- Descripción del método: Establece el asesor asociado al historial.

- Objetivo: Asignar un valor al asesor asociado al historial.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Asesor asesor.

public void setAsesor(Asesor asesor)

● getSector

- Descripción del método: Devuelve el sector del mercado asociado al historial.

- Objetivo: Obtener el sector del mercado asociado al historial.

- Ámbito: Público.

- Tipo de retorno: SectorDelMercado.

- Parámetros: Ninguno.

public SectorDelMercado getSector()

● setSector

- Descripción del método: Establece el sector del mercado asociado al historial.

- Objetivo: Asignar un valor al sector del mercado asociado al historial.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: SectorDelMercado sector.

public void setSector(SectorDelMercado sector)

**NOMBRE DE LA CLASE:**

● SectorDelMercado

**DESCRIPCIÓN:**

● Esta clase representa un sector del mercado con sus respectivas propiedades como id, nombre y descripción.

**PROPIEDADES DE LA CLASE:**

● id. Tipo: Int. Es requerido: Sí.

● nombre. Tipo: String. Es requerido: Sí.

● descripcion. Tipo: String. Es requerido: Sí.

**OPERACIONES (MÉTODOS) DE LA CLASE:**

● getNombre

- Descripción del método: Devuelve el nombre del sector del mercado.

- Objetivo: Obtener el nombre del sector del mercado.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getNombre()

● setNombre

- Descripción del método: Establece el nombre del sector del mercado.

- Objetivo: Asignar un valor al nombre del sector del mercado.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String nombre.

public void setNombre(String nombre)

● getDescripcion

- Descripción del método: Devuelve la descripción del sector del mercado.

- Objetivo: Obtener la descripción del sector del mercado.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getDescripcion()

● setDescripcion

- Descripción del método: Establece la descripción del sector del mercado.

- Objetivo: Asignar un valor a la descripción del sector del mercado.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String descripcion.

public void setDescripcion(String descripcion)

● toString

- Descripción del método: Devuelve una representación en formato String del sector del mercado.

- Objetivo: Obtener una representación en formato String del sector del mercado.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String toString()

**NOMBRE DE LA CLASE:**

● SectoresEmpresas

**DESCRIPCIÓN:**

● Esta clase representa la relación entre una empresa y un sector del mercado.

**PROPIEDADES DE LA CLASE:**

● id. Tipo: Int. Es requerido: Sí.

● empresa. Tipo: Empresa. Es requerido: Sí.

● sector. Tipo: SectorDelMercado. Es requerido: Sí.

**OPERACIONES (MÉTODOS) DE LA CLASE:**

● getEmpresa

- Descripción del método: Devuelve la empresa asociada.

- Objetivo: Obtener la empresa asociada.

- Ámbito: Público.

- Tipo de retorno: Empresa.

- Parámetros: Ninguno.

public Empresa getEmpresa()

● getSector

- Descripción del método: Devuelve el sector del mercado asociado.

- Objetivo: Obtener el sector del mercado asociado.

- Ámbito: Público.

- Tipo de retorno: SectorDelMercado.

- Parámetros: Ninguno.

public SectorDelMercado getSector()

**NOMBRE DE LA CLASE:**

● PaisesOperacion

**DESCRIPCIÓN:**

● Esta clase representa la relación entre una empresa y un país en el que opera.

**PROPIEDADES DE LA CLASE:**

● id. Tipo: Int. Es requerido: Sí.

● empresa. Tipo: Empresa. Es requerido: Sí.

● pais. Tipo: Pais. Es requerido: Sí.

**OPERACIONES (MÉTODOS) DE LA CLASE:**

● getEmpresa

- Descripción del método: Devuelve la empresa asociada.

- Objetivo: Obtener la empresa asociada.

- Ámbito: Público.

- Tipo de retorno: Empresa.

- Parámetros: Ninguno.

public Empresa getEmpresa()

● getPais

- Descripción del método: Devuelve el país asociado.

- Objetivo: Obtener el país asociado.

- Ámbito: Público.

- Tipo de retorno: Pais.

- Parámetros: Ninguno.

public Pais getPais()

**NOMBRE DE LA CLASE:**

● Pais

**DESCRIPCIÓN:**

● Esta clase representa un país con sus respectivas propiedades como id, nombre, producto interno bruto, población, capital y sede.

**PROPIEDADES DE LA CLASE:**

● id. Tipo: Int. Es requerido: Sí.

● nombre. Tipo: String. Es requerido: Sí.

● productoInternoBruto. Tipo: Double. Es requerido: Sí.

● poblacion. Tipo: Int. Es requerido: Sí.

● capital. Tipo: String. Es requerido: Sí.

● sede. Tipo: Sede. Es requerido: Sí.

**OPERACIONES (MÉTODOS) DE LA CLASE:**

● getNombre

- Descripción del método: Devuelve el nombre del país.

- Objetivo: Obtener el nombre del país.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getNombre()

● setNombre

- Descripción del método: Establece el nombre del país.

- Objetivo: Asignar un valor al nombre del país.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String nombre.

public void setNombre(String nombre)

● getProductoInternoBruto

- Descripción del método: Devuelve el producto interno bruto del país.

- Objetivo: Obtener el producto interno bruto del país.

- Ámbito: Público.

- Tipo de retorno: Double.

- Parámetros: Ninguno.

public Double getProductoInternoBruto()

● setProductoInternoBruto

- Descripción del método: Establece el producto interno bruto del país.

- Objetivo: Asignar un valor al producto interno bruto del país.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Double productoInternoBruto.

public void setProductoInternoBruto(Double productoInternoBruto)

● getPoblacion

- Descripción del método: Devuelve la población del país.

- Objetivo: Obtener la población del país.

- Ámbito: Público.

- Tipo de retorno: Int.

- Parámetros: Ninguno.

public Int getPoblacion()

● setPoblacion

- Descripción del método: Establece la población del país.

- Objetivo: Asignar un valor a la población del país.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Int poblacion.

public void setPoblacion(Int poblacion)

● getCapital

- Descripción del método: Devuelve la capital del país.

- Objetivo: Obtener la capital del país.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getCapital()

● setCapital

- Descripción del método: Establece la capital del país.

- Objetivo: Asignar un valor a la capital del país.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String capital.

public void setCapital(String capital)

● getSede

- Descripción del método: Devuelve la sede del país.

- Objetivo: Obtener la sede del país.

- Ámbito: Público.

- Tipo de retorno: Sede.

- Parámetros: Ninguno.

public Sede getSede()

● setSede

- Descripción del método: Establece la sede del país.

- Objetivo: Asignar un valor a la sede del país.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Sede sede.

public void setSede(Sede sede)

**NOMBRE DE LA CLASE:**

● Ciudad

**DESCRIPCIÓN:**

● Esta clase representa una ciudad con sus respectivas propiedades como id, nombre y sede.

**PROPIEDADES DE LA CLASE:**

● id. Tipo: Int. Es requerido: Sí.

● nombre. Tipo: String. Es requerido: Sí.

● idCiudad. Tipo: Int. Es requerido: Sí.

● sede. Tipo: Sede. Es requerido: Sí.

**OPERACIONES (MÉTODOS) DE LA CLASE:**

● getNombre

- Descripción del método: Devuelve el nombre de la ciudad.

- Objetivo: Obtener el nombre de la ciudad.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getNombre()

● setNombre

- Descripción del método: Establece el nombre de la ciudad.

- Objetivo: Asignar un valor al nombre de la ciudad.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String nombre.

public void setNombre(String nombre)

● getIdCiudad

- Descripción del método: Devuelve el ID de la ciudad.

- Objetivo: Obtener el ID de la ciudad.

- Ámbito: Público.

- Tipo de retorno: Int.

- Parámetros: Ninguno.

public Int getIdCiudad()

● setIdCiudad

- Descripción del método: Establece el ID de la ciudad.

- Objetivo: Asignar un valor al ID de la ciudad.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Int idCiudad.

public void setIdCiudad(Int idCiudad)

● getSede

- Descripción del método: Devuelve la sede asociada a la ciudad.

- Objetivo: Obtener la sede asociada a la ciudad.

- Ámbito: Público.

- Tipo de retorno: Sede.

- Parámetros: Ninguno.

public Sede getSede()

**NOMBRE DE LA CLASE:**

● Sede

**DESCRIPCIÓN:**

● Esta clase representa una sede con sus respectivas propiedades como id, nombre, empresa asociada, ciudad asociada y país asociado.

**PROPIEDADES DE LA CLASE:**

● id. Tipo: Int. Es requerido: Sí.

● nombre. Tipo: String. Es requerido: Sí.

● empresa. Tipo: Empresa. Es requerido: Sí.

● ciudad. Tipo: Ciudad. Es requerido: Sí.

● pais. Tipo: Pais. Es requerido: Sí.

**OPERACIONES (MÉTODOS) DE LA CLASE:**

● getNombre

- Descripción del método: Devuelve el nombre de la sede.

- Objetivo: Obtener el nombre de la sede.

- Ámbito: Público.

- Tipo de retorno: String.

- Parámetros: Ninguno.

public String getNombre()

● setNombre

- Descripción del método: Establece el nombre de la sede.

- Objetivo: Asignar un valor al nombre de la sede.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: String nombre.

public void setNombre(String nombre)

● getEmpresa

- Descripción del método: Devuelve la empresa asociada a la sede.

- Objetivo: Obtener la empresa asociada a la sede.

- Ámbito: Público.

- Tipo de retorno: Empresa.

- Parámetros: Ninguno.

public Empresa getEmpresa()

● setEmpresa

- Descripción del método: Establece la empresa asociada a la sede.

- Objetivo: Asignar un valor a la empresa asociada a la sede.

- Ámbito: Público.

- Tipo de retorno: Ninguno.

- Parámetros: Empresa empresa.

public void setEmpresa(Empresa empresa)

● getCiudad

- Descripción del método: Devuelve la ciudad asociada a la sede.

- Objetivo: Obtener la ciudad asociada a la sede.

- Ámbito: Público.

- Tipo de retorno: Ciudad.

- Parámetros: Ninguno.

public Ciudad getCiudad()

● getPais

- Descripción del método: Devuelve el país asociado a la sede.

- Objetivo: Obtener el país asociado a la sede.

- Ámbito: Público.

- Tipo de retorno: Pais.

- Parámetros: Ninguno.

public Pais getPais()

# RELACIONES ENTRE CLASES

En esta sección deben identificar las relaciones que existen entre las diferentes clases y describir cada una defendiendo y representando su nivel o grado de cardinalidad o multiplicidad, según sea cada caso:

Recordar que existen diferentes tipos de tipos de relaciones entre clases:

* Relación de asociación con cardinalidad
* Relación de Herencia
* Relación de Composición
* Relación de Agregación
* Relación de clase Asociación
* Relación de Usabilidad

Empresa "1" -- "\*" Vendedor: Trabaja en

Vendedor "1" \*-- "1" Vendedor: Recluta

Vendedor --|> Usuario

Asesor --|> Usuario

Asesor "1" \*-- "0..\*" Empresa

Empresa "1" --\* "1" Sede

Empresa "\*" --\* "1..\*" SectorDelMercado

(Empresa, SectorDelMercado) .. SectoresEmpresas

HistoricoAsesor "1" -- "1" Asesor

HistoricoAsesor "1" -- "1" Empresa

HistoricoAsesor "1" -- "1" SectorDelMercado

Asesor "\*" --\* "\*" SectorDelMercado

(Asesor, SectorDelMercado) .. AsesoresSectores

Empresa "\*" --\* "1..\*" Pais

(Empresa, Pais) .. PaisesOperacion

Ciudad "1" \*-- "1" Sede

Pais "1" \*-- "1" Sede

# MODELAR EN UML EL RESPECTIVO DIAGRAMA DE CLASE

# 

# DEL DIAGRAMA AL CÓDIGO UML

**@startuml**

**class** **Usuario** {

#**String:** codigo;

#**String:** nombre;

#**String:** direccion;

#**String:** nombreUsuario;

#**String:** clave;

#**String:** perfil;

+Usuario();

+getCodigo();

+setCodigo();

+getNombre();

+setNombre();

+getDireccion();

+setDireccion();

+getNombreUsuario();

+setNombreUsuario();

+getClave();

+setClave();

+getPerfil();

+setPerfil();

}

**class** **Vendedor** {

-**Int:** id;

-**Date:** fechaReclutamiento;

-**Int:** id\_lider;

-**Empresa:** empresa;

+getId();

+getFechaReclutamiento();

+setFechaReclutamiento();

+getId\_lider();

+setItem();

+getEmpresa();

+setEmpresa();

}

**class** **Asesor** {

-**Int:** id;

-**titulacion:** string

-**fechaInicioTrabajo:** date

+getId();

+getTitulacion();

+setTitulacion();

+getFechaTrabajo();

+setFechaTrabajo();

}

**class** **Empresa** {

-**Int:** id;

-**String:** nombre;

-**Date:** fechaIncorporacion;

-**Double:** facturacionAnual;

-**Int:** numeroVendedores;

-**Sede:**sede;

-List<Vendedor> vendedores;

+getId();

+getNombre();

+setFechaIncorporacion();

+getFacturacionAnual();

+setFacturacionAnual();

+getNumeroVendedores();

+setNumeroVendedores();

+getSede();

+getVendedores();

+toString();

}

**class** **AsesoresSectores** {

-**Int:** id;

-**Asesor:** asesor;

-**SectorDelMercado:** sector;

+getAsesor();

+setAsesor();

+getSector();

+getSector();

}

**class** **HistoricoAsesor** {

-**Int:** id;

-**Empresa:** empresa;

-**Asesor:** asesor;

-**SectorDelMercado:** sector;

-**Date:** fechaIncorporacion;

+getAsesor();

+setAsesor();

+getSector();

+getSector();

}

**class** **SectorDelMercado** {

-**Int:** id;

-**String:** nombre;

-**String:** descripcion;

+getNombre();

+setNombre();

+getDescripcion();

+setDescripcion();

+toString();

}

**class** **SectoresEmpresas** {

-**Int:** id;

-**Empresa:** empresa;

-**SectorDelMercado:** sector;

+getEmpresa();

+getsector();

}

**class** **PaisesOperacion** {

-**Int:** id;

-**Empresa:** empresa;

-**Pais:** pais;

+getEmpresa();

+getPais();

}

**class** **Pais** {

-**Int:** id;

-**String:** nombre;

-**Double:** productoInternoBruto;

-**Int:** poblacion;

-**String:** capital;

-**Sede:** sede;

+getNombre();

+setNombre();

+getProductoInternoBruto();

+setProductoInternoBruto();

+getPoblacion();

+setPoblacion();

+getCapital();

+setCapital();

+getSede();

+setSede(Sede sede);

}

**class** **Ciudad** {

-**Int:** id;

-**String:** nombre

-**In:** idCiudad;

-**Sede:** sede;

+getNombre();

+setNombre();

+getIdCiudad();

+setIdCiudad();

+getSede();

}

**class** **Sede** {

-**Int:** id;

-**String:** nombre;

-**Empresa:** empresa;

-**Ciudad:** ciudad;

-**Pais:** pais;

+getNombre();

+setNombre(String nombre);

+getEmpresa();

+setEmpresa();

+getCiudad();

+getPais();

}

Empresa "1" -- "\*" **Vendedor:** Trabaja en

Vendedor "1" \*-- "1" **Vendedor:** Recluta

Vendedor --|> Usuario

Asesor --|> Usuario

Asesor "1" \*-- "0..\*" Empresa

Empresa "1" --\* "1" Sede

Empresa "\*" --\* "1..\*" SectorDelMercado

(Empresa, SectorDelMercado) .. SectoresEmpresas

HistoricoAsesor "1" -- "1" Asesor

HistoricoAsesor "1" -- "1" Empresa

HistoricoAsesor "1" -- "1" SectorDelMercado

Asesor "\*" --\* "\*" SectorDelMercado

(Asesor, SectorDelMercado) .. AsesoresSectores

Empresa "\*" --\* "1..\*" Pais

(Empresa, Pais) .. PaisesOperacion

Ciudad "1" \*-- "1" Sede

Pais "1" \*-- "1" Sede

**@enduml**

# ENLACE DE GITHUB

https://github.com/egutierrez0323/EmmanuelGutierrezAct1Poo20241.git

# BIBLIOGRAFÍA

Arrieta, J. [John Carlos Arrieta Arrieta]. (2021, Marzo 16). EJEMPLO DE ANALISIS Y DISEÑO ORIENTADO A OBJETOS - PARTE 2 - DEL MODELO AL CODIGO JAVA [Video]. Recuperado de <https://www.youtube.com/watch?v=aKeMmGLVQb4&list=PLfB6iylhfbYtmfgYtxlRuGGzNR6fT-MUV&index=9>