

Enunciado

Desde hace mucho tiempo Ada no compra sus zapatos, botas, bolsos y otros artículos de piel en otro sitio que no sea "El tacón de oro". Conoce a los dueños de esta antigua zapatería desde hace tiempo y sabe que la nueva generación está muy interesada en expandir el negocio. Después de abrir varias sucursales, en diferentes puntos de la ciudad, y de otros lugares, opinan que el siguiente paso es la venta por Internet. Tienen muy clara su idea, poner los artículos de temporada (son los que están disponibles en almacén) a la venta, en forma de campañas. Mientras dure una campaña se podrán visualizar los artículos a la venta y hacer pedidos de los mismos, siempre y cuando se sea socio. El administrativo será el encargado de abrir y cerrar las campañas, y habrá un empleado de almacén que se encargue de servir los productos.

Descripción exacta del problema: "El tacón de oro"

Los usuarios del sistema navegan por la web para ver los artículos, zapatos, bolsos y complementos que se venden en la tienda. De los artículos nos interesa su nombre, descripción, material, color, precio y stock. De los zapatos nos interesa su número y el tipo. De los bolsos nos interesa su tipo (bandolera, mochila, fiesta). De los complementos (cinturones y guantes) su talla.

Los artículos se organizan por campanñas para cada temporada (primavera/verano y otoño/invierno) de cada año.

Los artículos son de fabricación propia, pero, opcionalmente, pueden venderse artículos de otras firmas. De las firmas nos interesa saber su nombre, CIF y domicilio fiscal. La venta de artículos de firma se realiza a través de proveedores, de forma que un proveedor puede llevar varios artículos de diferentes firmas, y una firma puede ser suministrada por más de un proveedor. Los artículos pertenecen a una firma solamente. De los proveedores debemos conocer su nombre, CIF, y domicilio fiscal.

Los usuarios pueden registrarse en el sitio web para hacerse socios. Cuando un usuario se hace debe proporcionar los siguiente datos: nombre completo, correo electrónico y dirección.

Los socios pueden hacer pedidos de los artículos. Un pedido está formado por un conjunto de detalles de pedido que son parejas formadas por artículo y la cantidad. De los pedidos interesa saber la fecha en la que se realizó y cuanto debe pagar el socio en total. El pago se hace a través tarjeta bancaria, cuando se va a pagar una entidad bancaria comprueba la validez de la tarjeta. De la tarjeta interesa conocer el número.

Las campañas son gestionadas por el administrativo de la tienda que se encargará de dar de baja la campaña anterior y dar de alta la nueva siempre que no haya ningún pedido pendiente de cumplimentar.

Existe un empleado de almacén que revisa los pedidos a diario y los cumplimenta. Esto consiste en recopilar los artículos que aparecen en el pedido y empaquetarlos. Cuando el paquete está listo se pasa al almacén a la espera de ser repartido. Del reparto se encarga una empresa de transportes que tiene varias rutas preestablecidas. Según el destino del paquete (la dirección del socio) se asigna a una u otra ruta. De la empresa de transportes se debe conocer su nombre, CIF y domicilio fiscal. Las rutas tienen un área de influencia que determina los destinos, y unos días de reparto asignados. Se debe conocer la fecha en la que se reparte el pedido. Si se produce alguna incidencia durante el reparto de algún pedido se almacena la fecha en la que se ha producido y una descripción.

Los socios pueden visualizar sus pedidos y cancelarlos siempre y cuando no hayan sido cumplimentados por el empleado de almacén. Así mismo puede modificar sus datos personales.

1. Crear la tabla de sustantivos y categorizarlos.

Tabla de sustantivos	
Usuarios -> Socios	Entidad Externa o rol
Artículos	Entidad Externa o rol
Zapatos	Cosa
Bolsos	Cosa
Complementos	Cosa
Nombre	Atributo
Descripcion	Atributo
Material	Atributo
Color	Atributo
Precio	Atributo
Stock	Atributo
Numero	Atributo
Tipo	Atributo
Bandolera	Valor de atributo
Mochila	Valor de atributo
Fiesta	Valor de atributo
Cinturones	Valor de atributo
Guantes	Valor de atributo
Talla	Atributo
Campaña	Cosa
Primavera/verano	Valor de atributo
Otono/invierno	Valor de atributo
Anio	Atributo
Firmas	Entidad Externa o rol

CIF	Atributo
Domicilio fiscal	Atributo
Proveedores	Entidad Externa o rol
Socios	Entidad externa o rol
Correo electronico	Atributo
Direccion	Atributo
Pedidos	Entidad Externa o rol
Detalles	Cosa
Cantidad	Atributo
Fecha	Atributo
Pago	Cosa
Tarjeta bancaria	Cosa
Entidad bancaria	Entidad Externa o rol
Numero	Atributo
Administrativo	Entidad Externa o rol
Empleado de almacen	Entidad Externa o rol
Almacen	Entidad Externa o rol
Reparto	Cosa
Empresa de transportes	Entidad Externa o rol
Rutas	Cosa
Destino	Atributo
Area de influencia	Atributo
Dias de reparto	Atributo
Incidencia	Cosa

2. Seleccionar las clases.

Basándonos en los siguientes criterios:

1. La información de la clase es necesaria para que el sistema funcione.
2. La clase posee un conjunto de atributos que podemos encontrar en cualquier ocurrencia de sus objetos. Si sólo aparece un atributo normalmente se rechazará y será añadido como atributo de otra clase.
3. La clase tiene un conjunto de operaciones identificables que pueden cambiar el valor de sus atributos y son comunes en cualquiera de sus objetos.
4. Es una entidad externa que consume o produce información esencial para la producción de cualquier solución en el sistema.

Teniendo en cuenta que se deben cumplir la mayoría de estos criterios, podemos seleccionar los siguientes sustantivos como objeto:

Clase candidata	Criterios que cumple
Artículos	1, 2, 3
Zapatos	1, 2, 3
Bolsos	1, 2, 3
Complementos	1, 2, 3
Campaña	1, 2, 3
Firmas	2, 3, 4
Proveedores	2, 3, 4
Socios	2, 3, 4
Pedidos	1, 2, 3
Detalles	1, 2, 3
Pago	1, 2, 3
Tarjeta bancaria	1, 2, 3
Entidad bancaria	2, 4
Administrativo	1, 2, 3, 4
Empleado de almacén	1, 2, 3, 4
Almacén	1, 2, 3
Paquetes	2, 3
Reparto	2, 3
Empresa de transportes	2, 3, 4
Rutas	2, 3
Incidencia	2, 3

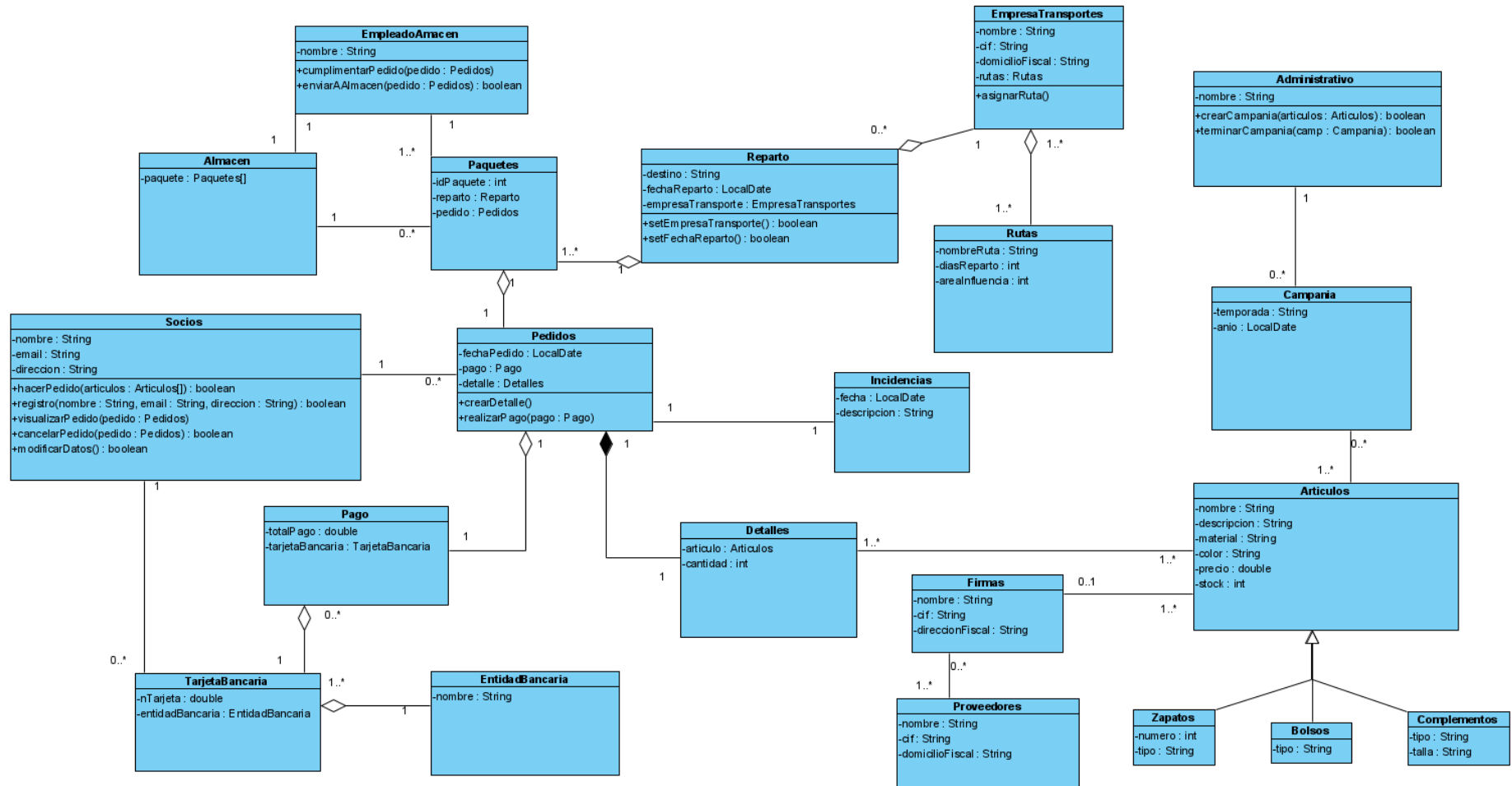
3. Obtención de los atributos de los objetos.

Clase candidata	Criterios que cumple
Artículos	nombre, descripcion, material, color, precio, stock
Zapatos	numero, tipo
Bolsos	tipo
Complementos	tipo, talla
Campaña	temporada, anio
Firmas	nombre, cif, domicilioFiscal
Proveedores	nombre, cif, domicilioFiscal
Socios	nombre, email, dirección
Pedidos	fechaPedido, detalles, pago
Detalles	articulo, cantidad
Pago	totalPago, tarjetaBancaria
Tarjeta bancaria	nTarjeta, entidadBancaria
Entidad bancaria	nombre
Administrativo	nombre
Empleado de almacen	nombre
Almacen	paquete
Paquetes	idPaquete, pedido, reparto
Reparto	destino, fechaReparto, empresaTransporte
Empresa de transportes	nombre, cif, domicilio, ruta
Rutas	nombreRuta, diasReparto, areaInfluencia
Incidencia	fecha, descripcion

4. Obtención de métodos

Clase candidata	Criterios que cumple
Socios	hacerPedido(articulos : Articulo[]) : boolean registro(nombre : String, email : String, direccion: String) : boolean visualizarPedido(pedido : Pedidos) : void cancelarPedido(pedido: Pedidos) : boolean modificarDatos() : boolean
Paquetes	asignarReparto() : void
Pedidos	crearDetalle() : void realizarPago(pago : Pago) : boolean
Administrativo	crearCampania(artículos : Articulos[]) : boolean terminarCampania(campania : Campania) : boolean
Empleado de almacen	cumplimentarPedido() : void enviarAAlmacen(Pedidos: pedido) : void
Almacen	
Reparto	asignarEmpresaTransporte() : boolean
Empresa de transportes	asignarRuta() : void
Rutas	
Incidencia	generarIncidencia() : void

5. Obtención de las relaciones

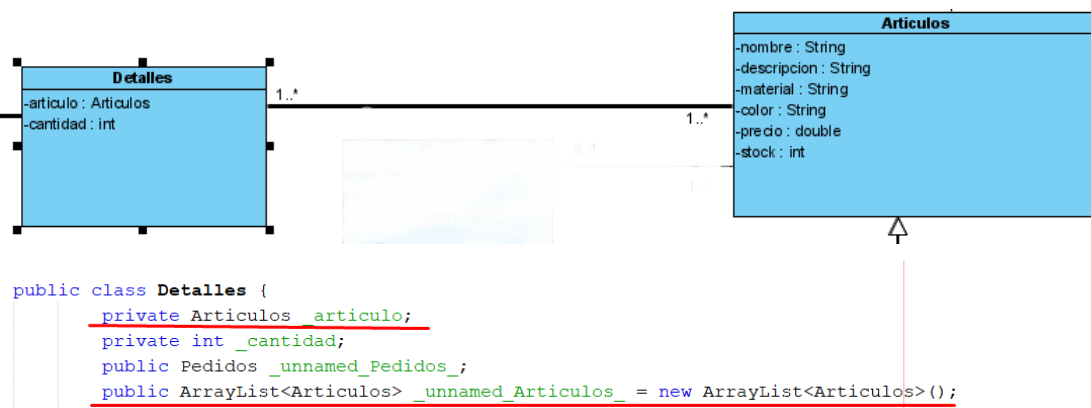


6. Setters y Getters y constructores

Setters, Getters y Constructores		Empleado de almacen	EmpleadoAlmacen(p : Pedido) setNombre(), getNombre()
Artículos	Articulos() setNombre(), getNombre(), setDescripcion(), getDescripcion(), setMaterial(), getMaterial(), setColor(), getColor(), setStock(), getStock()	Almacen	Almacen() setPaquete(), getPaquete()
Zapatos	Zapatos() setNumero(), getNumero(), setTipo(), getTipo()	Paquetes	Paquetes(r: Reparto, p: Pedido) setIdPaquete(), getIdPaquete(), setReparto(), getReparto(), setPedido(), getPedido()
Bolsos	Bolsos() setTipo(), getTipo()	Reparto	Reparto(et : EmpresaTransportes) setDestino(), getDestino(), setFechaReparto(), getFechaReparto(), setEmpresaTransportes(), getEmpresaTransportes()
Complementos	Complementos() setTipo(), getTipo(), setTalla(), getTalla()	Empresa de transportes	EmpresaTransportes(r : Rutas[]) setNombre(), getNombre(), setCif(), getCif(), setDomicilioFiscal(), getDomicilioFiscal(), setRuta(), getRuta()
Campaña	Campania() setTemporada(), getTemporada(), setAnio(), getAnio()	Rutas	Rutas() setNombreRuta(), getNombreRuta(), setAreaInfluencia(), getAreaInfluencia(), setDiasReparto(), getDiasReparto()
Firmas	Firmas(p : Proveedores) setNombre(), getNombre(), setCif(), getCif(), setDomicilioFiscal(), getDomicilioFiscal()	Incidencias	Incidencias() setFecha(), getFecha(), setDescripcion(), getDescripcion()
Proveedores	Proveedores() setNombre(), getNombre(), setCif(), getCif(), setDomicilioFiscal(), getDomicilioFiscal()		
Socios	Socios() setNombre(), getNombre(), setEmail(), getEmail(), setDireccion(), getDireccion()		
Pedidos	Pedidos(Detalles d, Pago p) setFechaPedido(), getFechaPedido()		
Detalles	Detalles() setArticulo(), getArticulo(), setCantidad(), getCantidad()		
Pago	Pago(t : TarjetaBancaria) setTotalPago(), getTotalPago(), setTarjetaBancaria(), getTarjetaBancaria()		
Tarjeta bancaria	TarjetaBancaria(eb: Entidad Bancaria) setTarjeta(), getTarjeta(), setEntidadBancaria(), getEntidadBancaria(),		
Entidad bancaria	EntidadBancaria() setNombre(), getNombre()		
Administrativo	Administrativo() setNombre(), getNombre()		

7. Refinamiento

- En un posible refinamiento del programa, podríamos incluir una clase llamada Empleado, del que a su vez hereden clases de distintos tipos o roles de empleado, para lo cual, "Empleado de Almacén" y "Administrativo" pasarían clases heredadas de la misma, con funciones propias de cada rol. Los atributos comunes de la clase Empleado podrían ser los datos personales de cada empleado, sueldo, etc.
- Los distintos tipos de artículos, como son: bandoleras, mochila, fiesta (para bolsos), podrían ser variables de tipo enumeradas dentro de la clase Bolsos.
- Los distintos tipos de artículos, como son: guantes y cinturones (para complementos), podrían ser variables de tipo enumeradas de la clase Complementos.
- Los dos distintos tipos de temporada: primavera/verano y otoño/invierno podrían ser de tipo enumerado dentro de la clase Campañas.
- En varias clases deberíamos generar métodos que puedan añadir otros objetos con los que pueden trabajar. Por ejemplo:
 - o en la clase Proveedores deberíamos generar métodos para añadir o eliminar las Firmas con las que trabajan.
 - o En la clase Detalles deberíamos generar un método para añadir o eliminar artículos.
 - o En la clase Socios deberíamos tener un método para generar y asignar a cada socio diferentes tarjetas de crédito.
 - o En la clase EmpleadoAlmacen podríamos generar un método que se encargue también de eliminar pedidos.
 - o En la clase Pedidos, además deberíamos tener un método generarPago, el cual maneje un objeto de tipo Pago asignándole la cantidad total a partir del objeto Detalle. También es importante crear un método que genere Incidencias.
- Los distintos objetos de tipo Paquete deben tener un atributo llamado estado, que controle las distintas fases de envío del propio paquete: "En Almacén", "En Reparto", etc...
- En el diagrama he incluido objetos de otras clases como atributos. Esto genera código redundante si ambas clases están relacionadas en el propio diagrama. En un posible refinamiento, habría que eliminar esos atributos, ya que al realizar la implementación automática del código se van a crear igualmente. Un ejemplo es la clase Detalle, la cual tiene un atributo Artículo:



8. Documentación

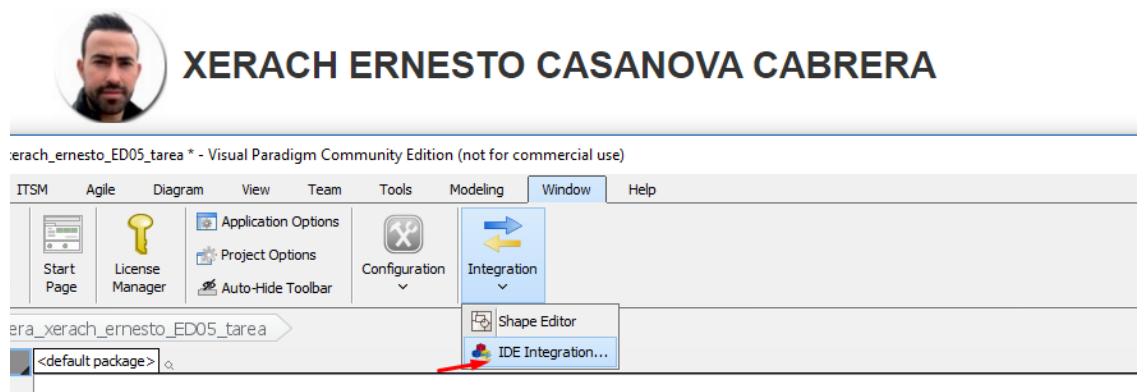
- **Clase Articulos:** Es una superclase de la cual heredan los distintos tipos de artículos: zapatos, bolsos y complementos, contiene los atributos comunes de dichas clases.

- **Clase Zapatos:** clase que hereda de Articulos, contiene objetos de tipo zapatos con sus atributos propios además de los heredados de la clase Articulos.
- **Clase Bolsos:** clase que hereda de artículos, contiene objetos de tipo Bolsos con sus atributos propios además de los heredados de la clase Articulos.
- **Clase Complementos:** clase que hereda de Articulos, contiene objetos de tipo Complementos con sus atributos propios además de los heredados de la clase Articulos.
- **Clase Camania:** Clase que genera objetos de tipo Campania, encargados de dividir los artículos de la tienda en distintas campañas (Primavera/Verano y Otoño/Invierno), ordenadas por temporada y año.
- **Clase Firmas:** Clase que genera objetos de tipo Firmas, de las cuales se obtienen los datos de las distintas firmas que trabajan con la tienda a través de la clase proveedores. Un objeto de tipo firmas estará relacionado con uno o más Proveedores y con cero o muchos Articulos.
- **Clase Proveedores:** Clase que genera objetos de tipo Proveedores, de los cuales se obtienen los datos de los distintos proveedores con los que trabaja la tienda. A su vez estos Proveedores estarán relacionados con uno o más objetos de tipo Firmas.
- **Clase Socios:** Clase que genera objetos de tipo Socios para los distintos usuarios que compran en la web. Con ella se podrán registrar nuevos socios con sus datos personales, los cuales se podrán modificar a través del método correspondiente, también se podrán realizar pedidos a través de un método que trabajará con un objeto de tipo Pedidos, al cual se le podrán asignar distintos Articulos. El usuario podrá visualizar sus pedidos o incluso cancelarlos a través de los métodos correspondientes.
- **Clase Pedidos:** Clase que genera objetos de tipo Pedidos, compuestos a su vez por un objeto de tipo Detalles y un método para generarlo, el cual se encarga de relacionar el objeto de tipo Pedido con los distintos Articulos de la tienda, un objeto de tipo Pago y un objeto de tipo Incidencia, así como una fecha de pedido.
- **Clase Detalles:** Clase que genera un objeto de tipo Detalles, cada objeto de tipo Detalle forma parte de un pedido y está compuesto de uno o varios objetos de tipo Articulo y una cantidad asignada para cada uno de los Articulos.
- **Clase Pago:** Clase que genera un objeto de tipo Pago, el cual irá asignado a un objeto de tipo Pedido, el cual tiene asignado un objeto de tipo TarjetaBancaria.
- **Clase TarjetaBancaria:** Clase que genera objetos de tipo TarjetaBacaria, compuestas por un número de tarjeta, un objeto de EntidadBancaria a la que pertenece la tarjeta y un objeto de tipo Socio al que está asignada la tarjeta.
- **Clase Administrativo.** Clase que genera objetos de tipo Administrativo, el cual podrá manejar objetos de tipo Camapania. Con este objeto se podrán modificar los datos del administrativo así como generar nuevos objetos de tipo Campania o finalizarlos.
- **Clase EmpleadoAlmacen:** Clase que genera objetos de tipo EmpleadoAlmacen, con este objeto se podrán modificar sus propios datos personales, así como crear objetos de tipo Paquete y asignarlos al objeto de tipo Almacen a través de sus métodos correspondientes.
- **Clase Almacen:** Clase que genera objetos de tipo Almacen, que irá almacenando objetos de tipo Paquete.
- **Clase Paquetes:** Clase que genera objetos de tipo Paquetes, que están formados por un objeto de tipo Pedido y otro objeto de tipo Reparto.

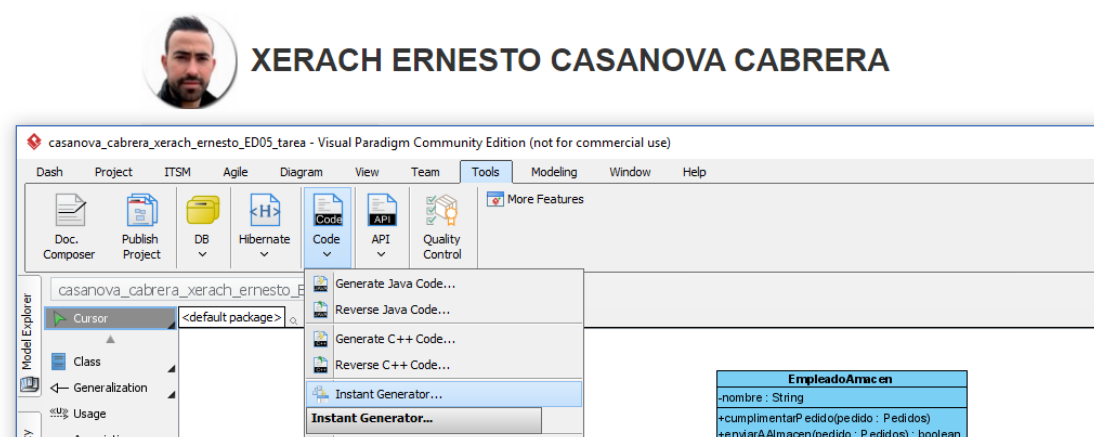
- **Clase Reparto:** Clase que genera objetos de tipo Reparto, que pueden ser asignados a objetos de tipo Pedido. Están formados por objetos de tipo EmpresaTransporte (encargado de asignar la ruta a ese reparto) y el atributo FechaReparto.
- **Clase EmpresaTransportes:** Clase que genera objetos de tipo EmpresaTransportes. Este objeto contiene objetos de tipo Ruta y se puede asignar a objetos de tipo Reparto, con los que trabajarán los distintos Paquetes. Además podremos modificar los datos fiscales de cada empresa de transporte.
- **Clase Rutas:** Clase que genera objetos de tipo Rutas, los cuales son asignados a los distintos objetos EmpresaDeTransporte. Estas rutas tendrán como atributos díasDeReparto y areaDeInfluencia, estrechamente relacionada con las distintas empresas de transporte.
- **Clase Incidencias:** Clase que genera objetos de tipo Incidencia, los cuales pueden ser relacionado con los objetos de tipo Pedido.

9. Generación automática del código

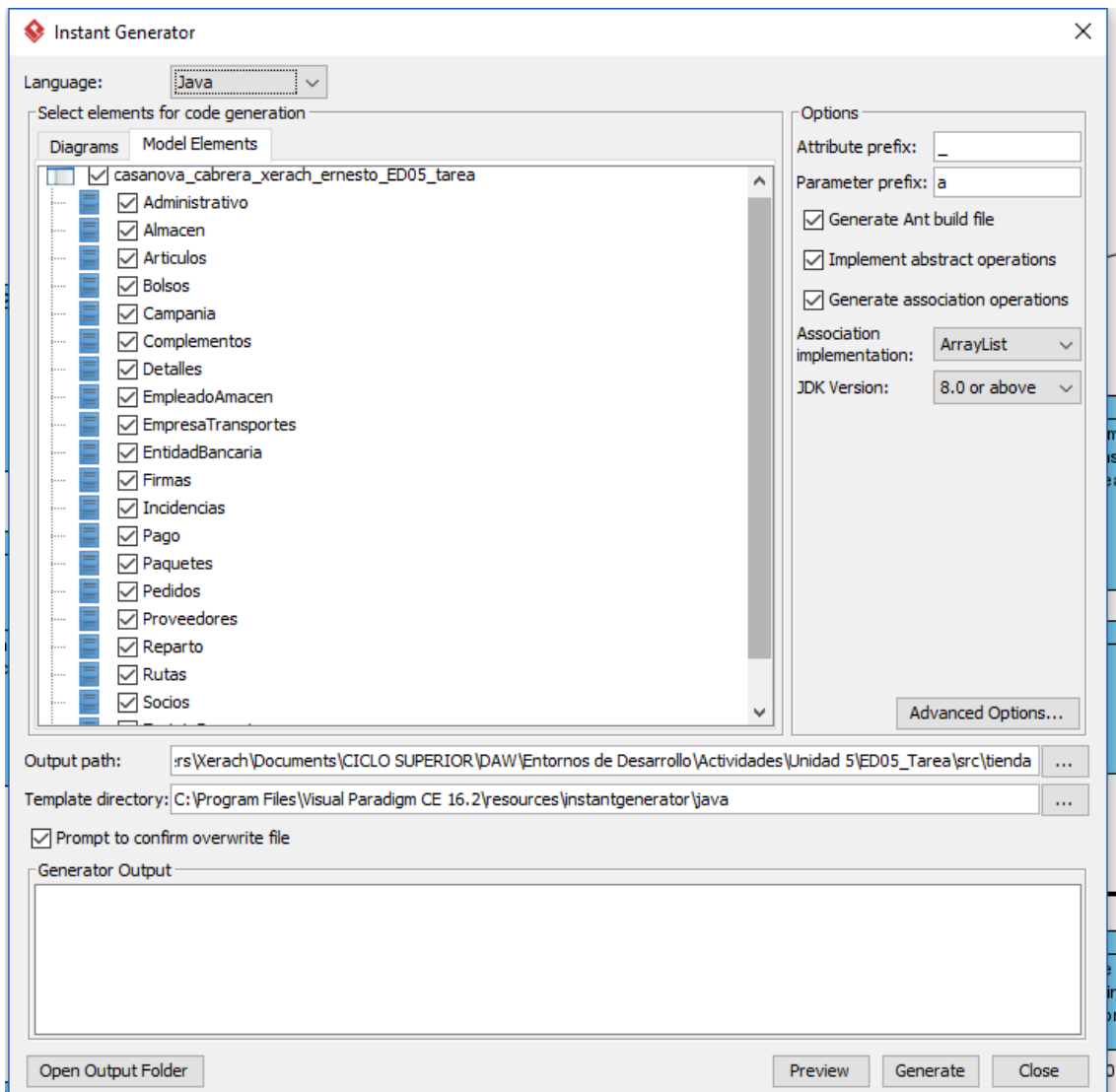
El primer paso es integrar Visual Paradigm en Netbeans, para ello me dirijo a la pestaña Window/IDE Integration. En la ventana que se abre, puedo seleccionar la integración en Netbeans y se descarga el plugin para este IDE.



Igualmente, al carecer de licencia no podemos utilizar bien la actualización. En su lugar, genero el código con la herramienta Instant Generator que proporciona Visual Paradigm



En la siguiente pantalla elegimos los elementos del diagrama de los cuales generaremos el código automáticamente.



Ejecuto NetBeans y genero un nuevo proyecto en el que puedo guardar los archivos .java generados para visualizarlos.



XERACH ERNESTO CASANOVA CABRERA

