

## TEMA 5-A03: Ferramentas para o traballo con diagramas de clases.

### Índice

---

<b>1.A03. Ferramentas para o traballo con diagramas de clases.....</b>	<b>2</b>
1.1Introdución .....	2
1.2Actividade .....	
Introdución .....	2
Criterios para a selección dunha ferramenta UML .....	2
Comparativas .....	2
Algunhas ferramentas Open Source.....	3
Ferramentas gratuítas.....	3
Algunhas ferramentas propietario.....	3
NetBeans. Módulo UML.....	4
Instalación .....	4
Xeración de código a partir de diagramas de clases.....	7
Xeración de diagramas de clase a partir de código (enxeñería inversa) .....	8
Visual Paradigm.....	10
Instalación de Visual Paradigm Community Edition .....	10
Xeración de código a partir de diagramas de clases.....	12
Xeración de diagramas de clase a partir de código (enxeñería inversa) .....	15
1.3Tarefas .....	17
1.3.1Tarefa 1. Instalación e manexo do módulo UML de NetBeans .....	17
1.3.2Tarefa 2. Instalación e manexo de Visual Paradigm .....	18

# 1. A03. Ferramentas para o traballo con diagramas de clases.

---

## 1.1 Introducción

Na actividade que nos ocupa aprenderanse os seguintes conceptos e manexo de destrezas:

- Instalar e manexar ferramentas para o traballo con diagramas UML.
- Empregar ferramentas software para xerar código fonte a partir de diagramas de clases e facer enxeñería inversa.

## 1.2 Actividade

### Introdución

Inda que o traballo con diagramas UML é perfectamente posible sen o emprego de ferramentas software, o mais habitual é empregar aplicacións que nos faciliten a representación de ditos diagramas e que incorporen a maiores unha serie de funcionalidades para o traballo con eles como poden ser a xeración automática de código a partir dos propios diagramas ou incluso a posibilidade de facer “enxeñería inversa” a partir do noso código, e dicir, xerar diagramas UML a partir de código fonte do noso proxecto.

### Criterios para a selección dunha ferramenta UML

Aspectos a ter en conta para elixir unha ferramenta que permita elaborar e traballar con modelos UML:

- Licenza e prezo.
- Plataforma sobre a que traballa.
- Tipo de diagramas que permite.
- Linguaxes de programación que permita asociar ao modelo e por tanto que poida xerar código a partir de diagramas.
- Posibilidade de xerar automaticamente documentación.
- Posibilidade de enxeñería inversa, é dicir, xerar diagramas a partir de código.
- Facilitade de navegación no modelo.
- Posibilidades de exportación do modelo.
- Interface de comunicación co usuario.

Existen centos de ferramentas UML independentes con licenza propietaria ou libre e moitos contornos de desenvolvemento que poden utilizar ferramentas UML.

### Comparativas

Relación de sitios web con comparativas entre ferramentas UML:

[http://en.wikipedia.org/wiki/Comparison\\_of\\_Unified\\_Modeling\\_Language\\_tools](http://en.wikipedia.org/wiki/Comparison_of_Unified_Modeling_Language_tools)

### Algunhas ferramentas Open Source



Dia (<http://dia-installer.de/index.html.en>)



Umbrello (<http://uml.sourceforge.net/>)



ArgoUML (<http://argouml.tigris.org/>)



Modelio (<http://www.modelio.org/>)



NetBeans (<https://netbeans.org/features/uml/>) é un contorno de desenvolvemento que dispón de complemento para realizar diagramas UML.

Ferramentas gratuítas



Jdeveloper é un contorno de desenvolvemento integrado desenvolvido por Oracle que simplifica o desenvolvemento de aplicacións baseadas en Java de xeito que contempla cada paso do ciclo de vida dunha aplicación.

Algunhas ferramentas propietario



Visual Paradigm (<http://www.visual-paradigm.com/>) e una potentísima ferramenta que nos permite traballar con proxectos UML. Permite a integración con NetBeans.

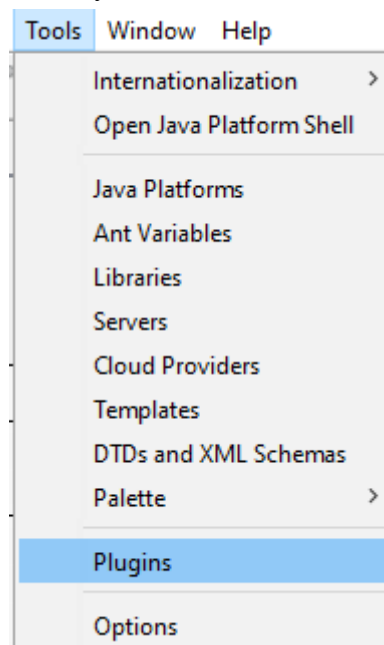
UModel (<http://www.altova.com/es/umodel.html>) defínese como unha ferramenta UML para deseñar software de forma visual. Deseña modelos de aplicacións con UML de forma visual e xera código Java, C++, C# o Visual Basic .NET, así como documentación de proxecto. Converte programas en diagramas UML mediante enxeñería inversa. Dispón dunha versión de proba de 30 días.

## NetBeans. Módulo UML

### Instalación

O contorno de desenvolvemento NetBeans permite instalar un módulo para o traballo con diagramas UML que ademais de aportarnos ferramentas gráficas para a representación dos distintos diagramas vai a darnos a posibilidade de xerar código a partir dos diagramas que teñamos feito e incluso poderemos facer “enxeñería inversa” a partir do noso código, e dicir, xerar diagramas UML a partir de código fonte do noso proxecto.

Imos ver como instalar o módulo easyUML en NetBeans 10,0

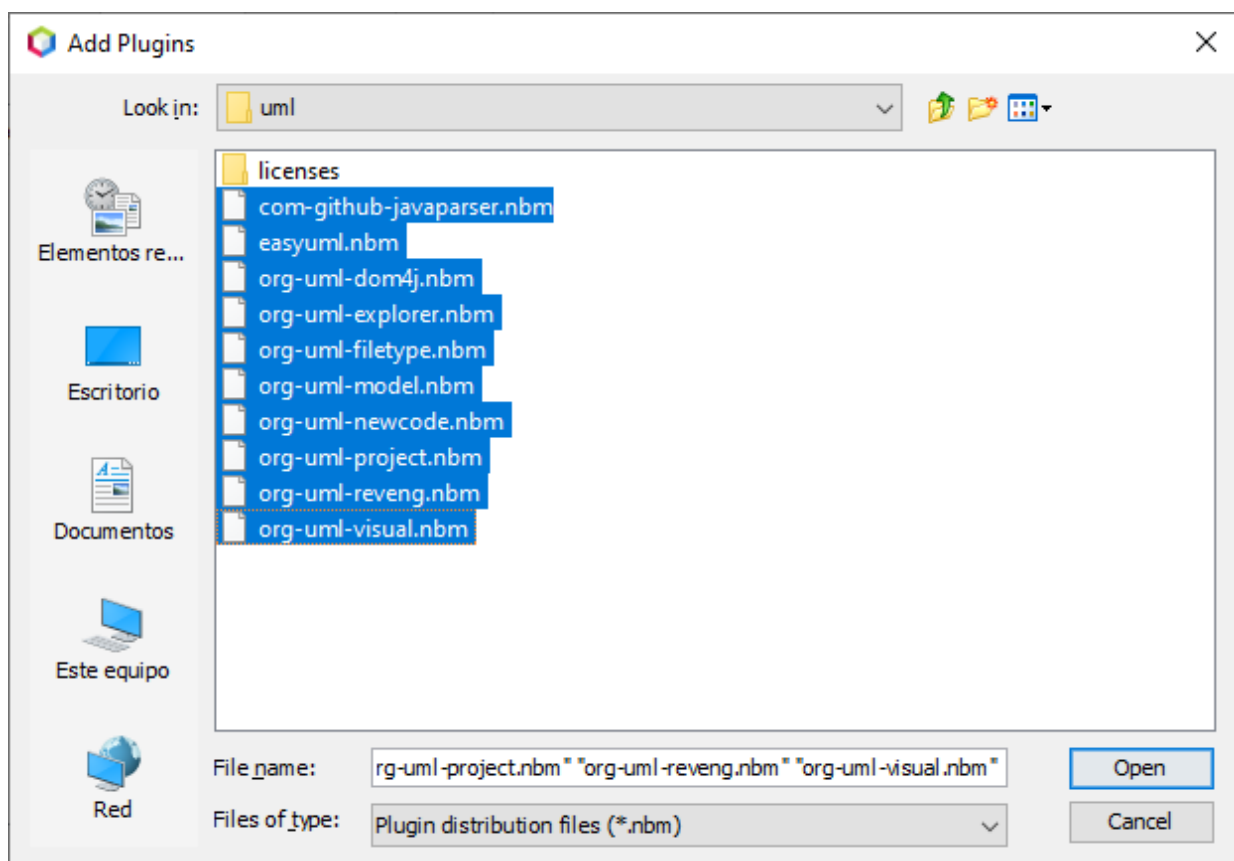
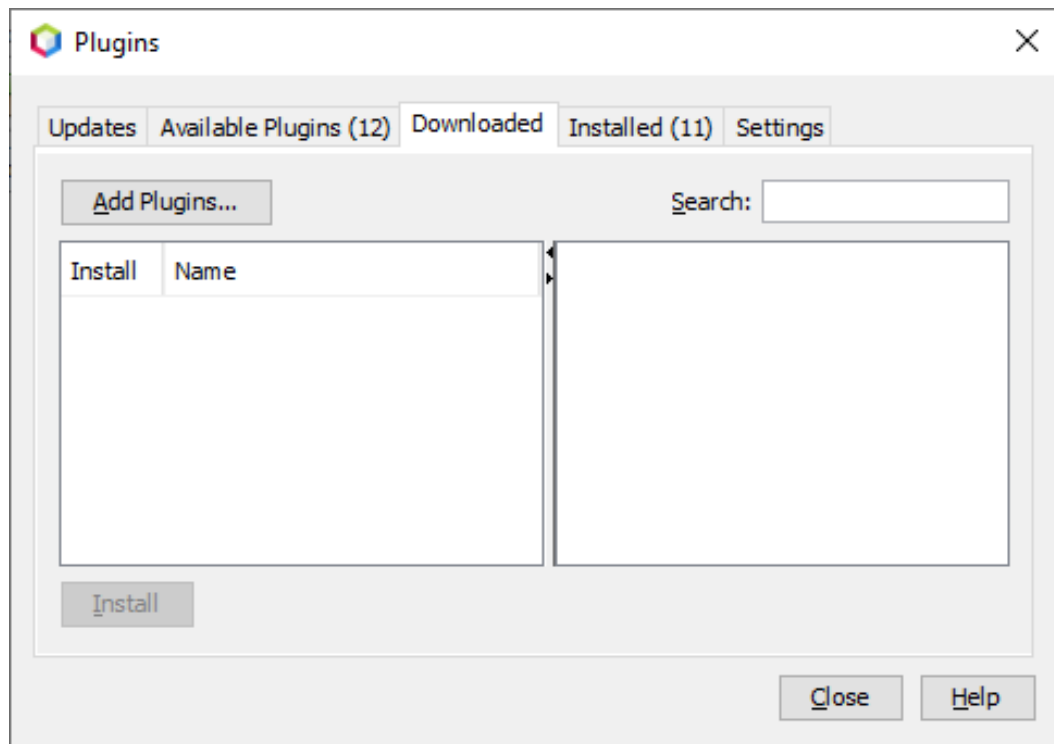


Iremos á opción *Plugins* do menú *Tools*:

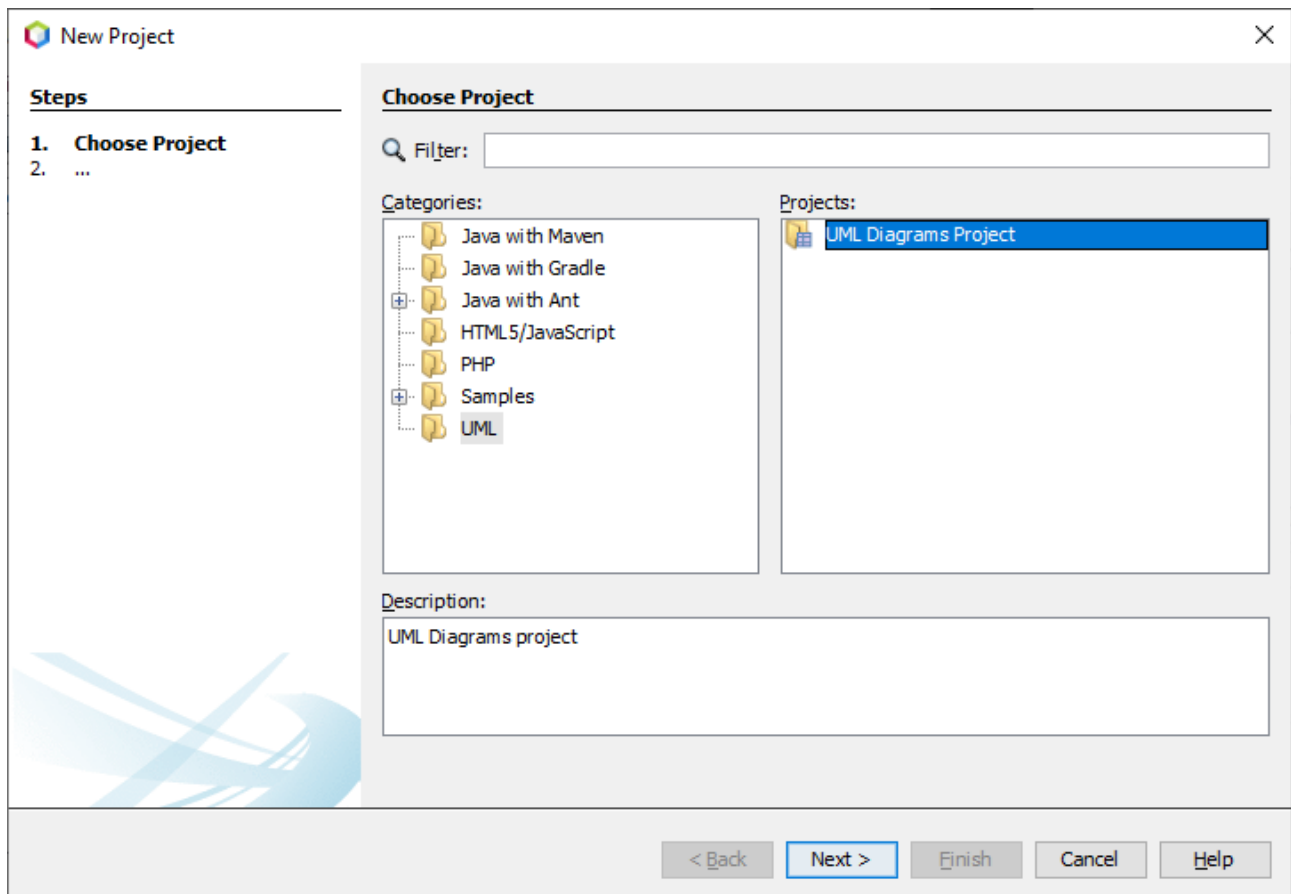
Non aparece o módulo easyUML polo que hai que descargalo dende

<http://plugins.netbeans.org/plugin/55435/easyuml>

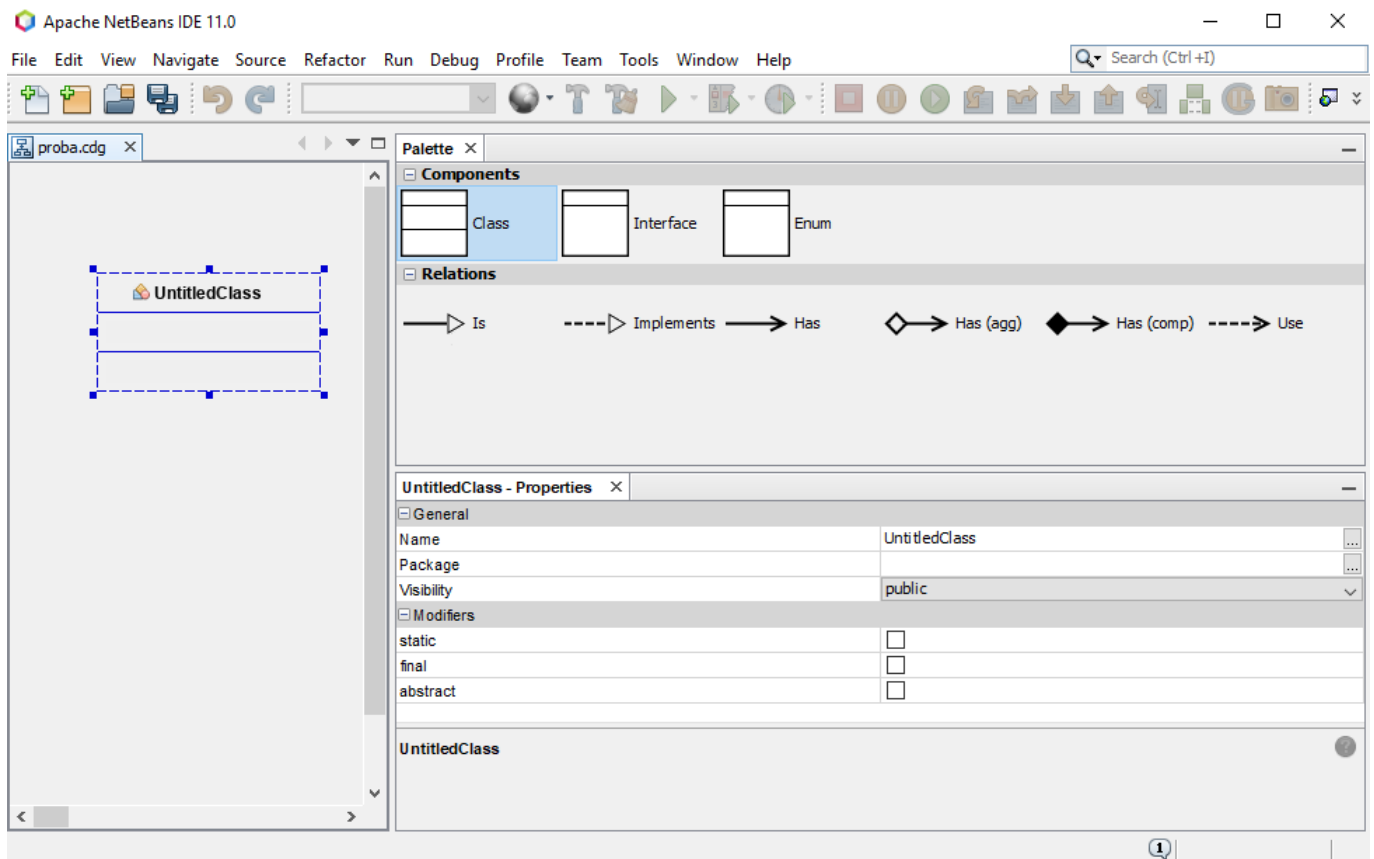
e engadilo na pestana de Downloaded e premer Add



Finalmente, despois da instalación comprobaremos que na opción de crear un proxecto novo nos parece a posibilidade de crear un proxecto UML:



E xa podemos empezar a traballar:

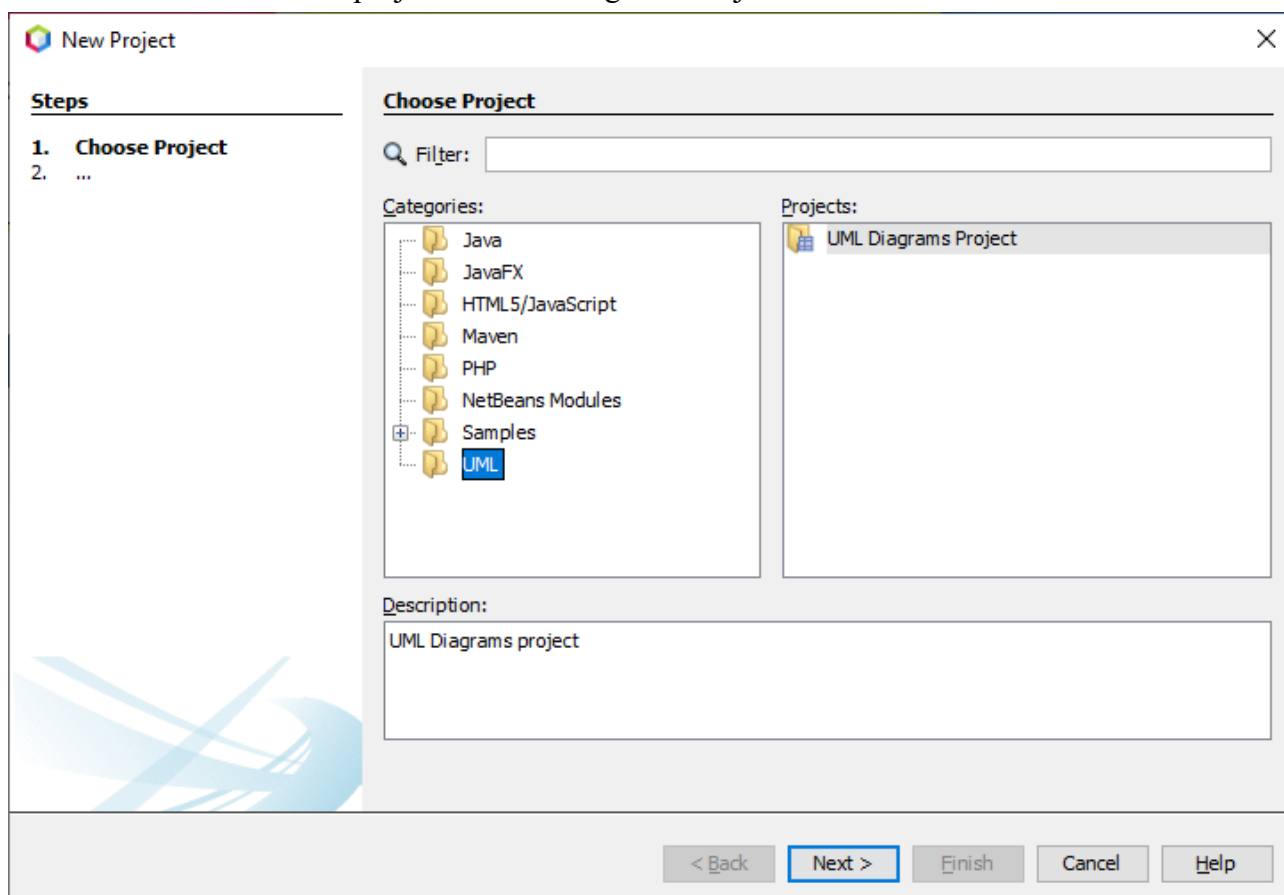


## Xeración de código a partir de diagramas de clases

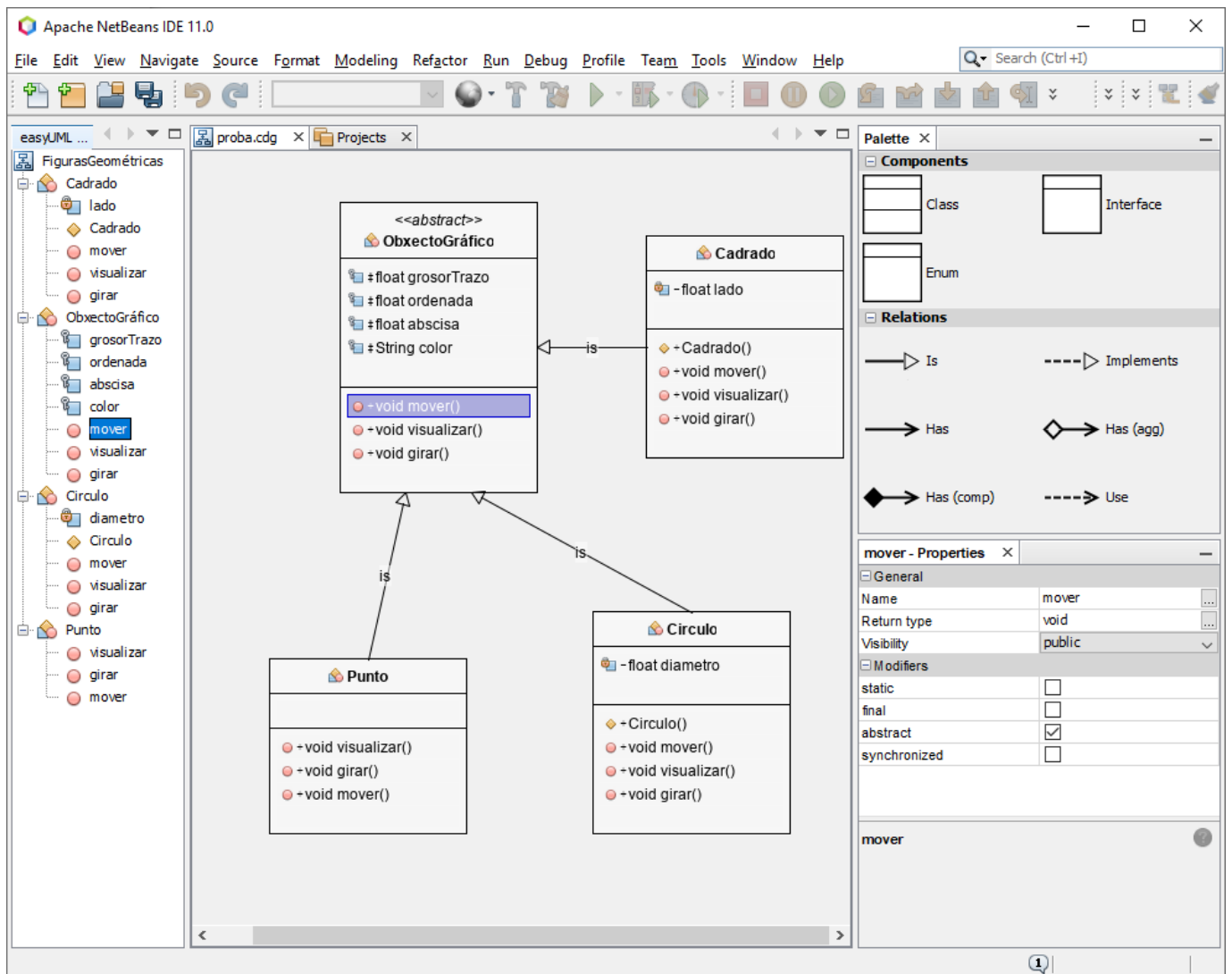
Imos ver como se pode xerar código co módulo UML de Netbeans a partir dun diagrama de clases existente.

En primeiro lugar temos que xerar un diagrama UML onde se van situar tódolos nosos diagramas UML que xeremos dos diferentes proxectos.

Netbeans-> New project-> UML Diagrams Project



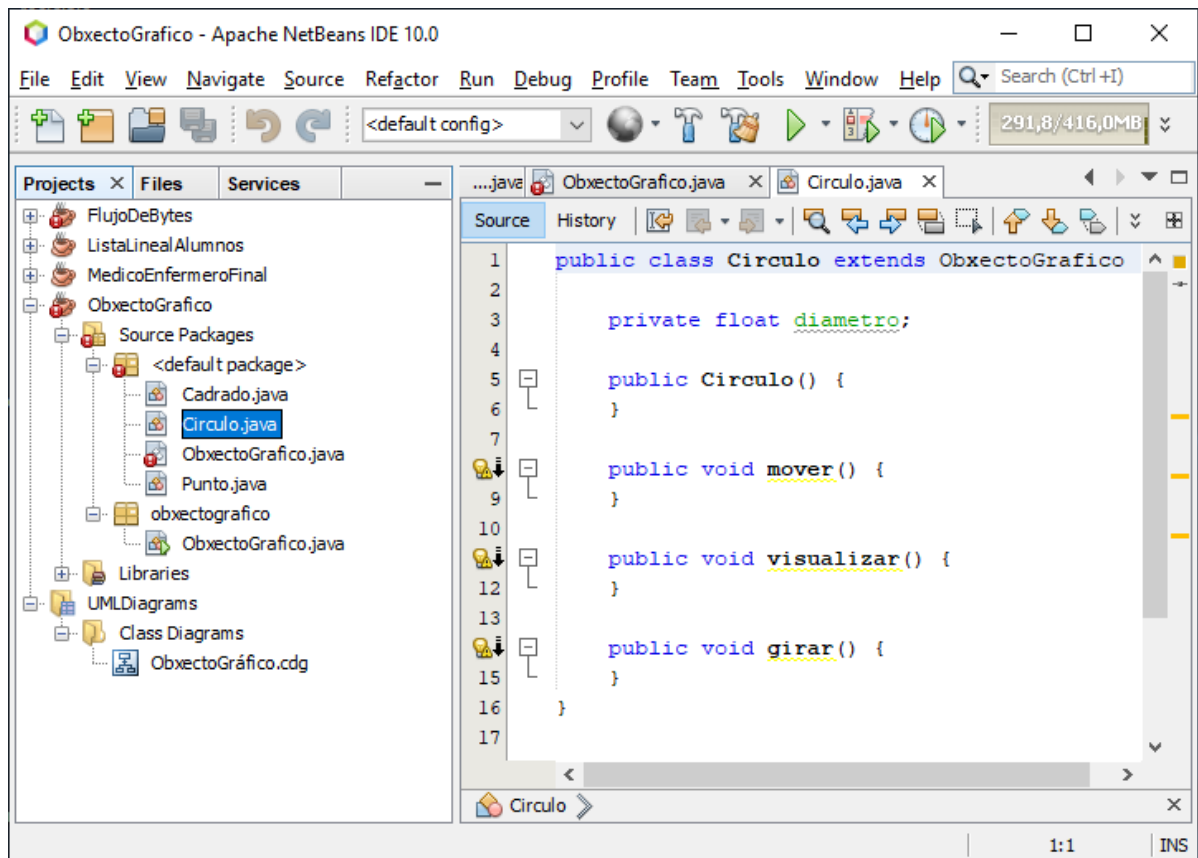
Despois imos xerando os diferentes elementos que compoñen o noso diagrama de clase. No noso caso estamos traballando cun obxecto xenérico chamado ObxectoGráfico que ten tres clases fillas: Cadrado, circulo e punto.



Agora temos que crear o proxecto Java onde imos situar o código que se xenera co diagrama de clases anterior. Para iso imos a File->New Project e lle ponemos de nome ObxectoGráfico

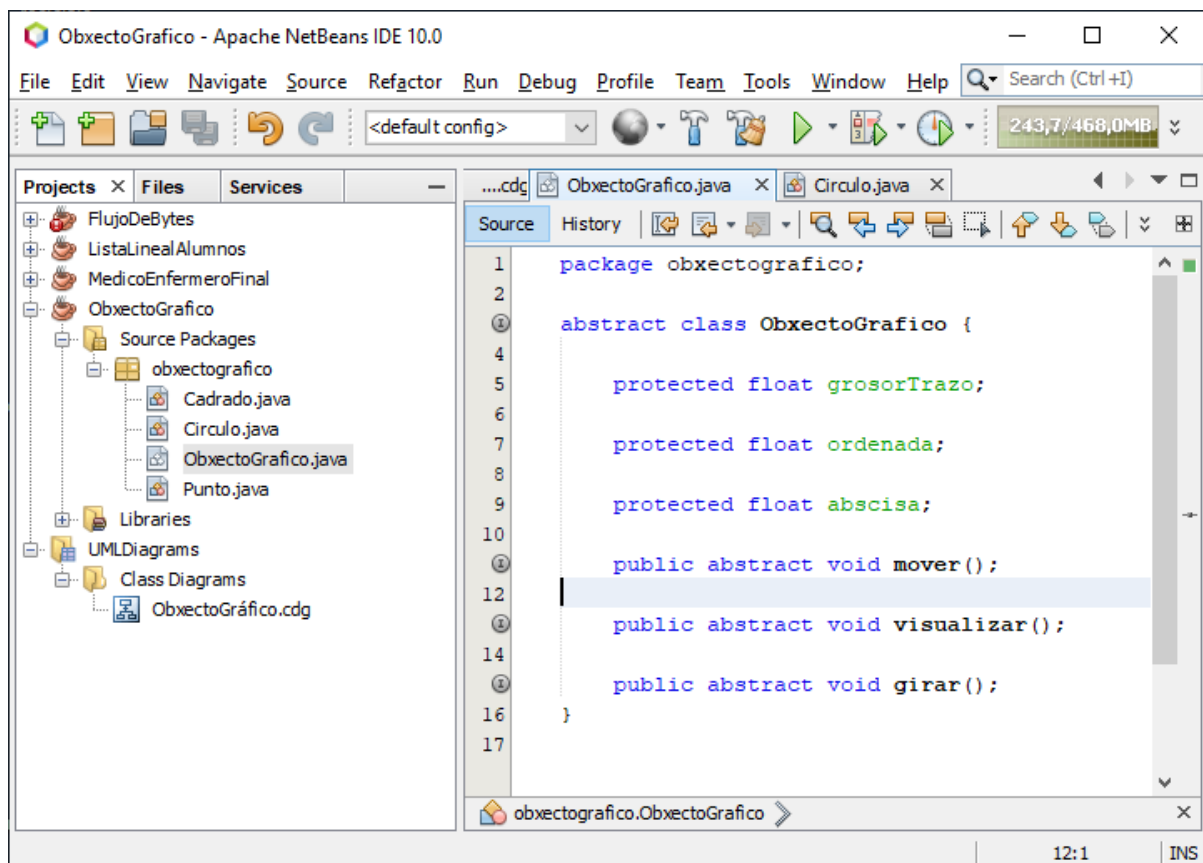
Situando o rato enriba do nome do diagrama de clases e premendo no botón dereito, seleccionaremos a opción de *EasyUML generate code..* Seleccionamos o nome do proxecto onde imos a xerar o código e pulsamos *Aceptar*.





Agora aparécenos as clases co código xeradas.

Debemos facer algunha corrección como eliminar o arquivo ObxectoGrafico.java do paquete Obxectografico, xa que é o arquivo baleiro que nos xera Java ó crear o proxecto. Eliminar o modificador de protexido na clase ObxectoGrafico e podemos mover tódalas clases a un paquete que tamén chamamos obxectografico





Realizar a Tarefa 1 consistente en instalar o módulo easyUML de Netbeans, crear diagramas e xerar código

## Visual Paradigm

### Instalación de Visual Paradigm Community Edition

Visual Paradigm é unha das ferramentas mais potentes para o traballo con diagramas UML. É software propietario pero dispón dunha versión de proba (FREE Trial) completamente funcional pero limita o seu uso a 30 días. Para a descarga da versión gratuíta iremos á páxina de descargas e seleccionaremos a *Community Edition* para o sistema operativo que estamos empregando (<http://www.visual-paradigm.com/download/community.jsp>):

Unha vez descargada, a executamos e comezará o proceso guiado de instalación que non ten ningunha dificultade:

## Download Visual Paradigm Community Edition

FREE for non-commercial use only

Version: 15.2  
Build number: 20190401

Download Visual Paradigm  
Community Edition

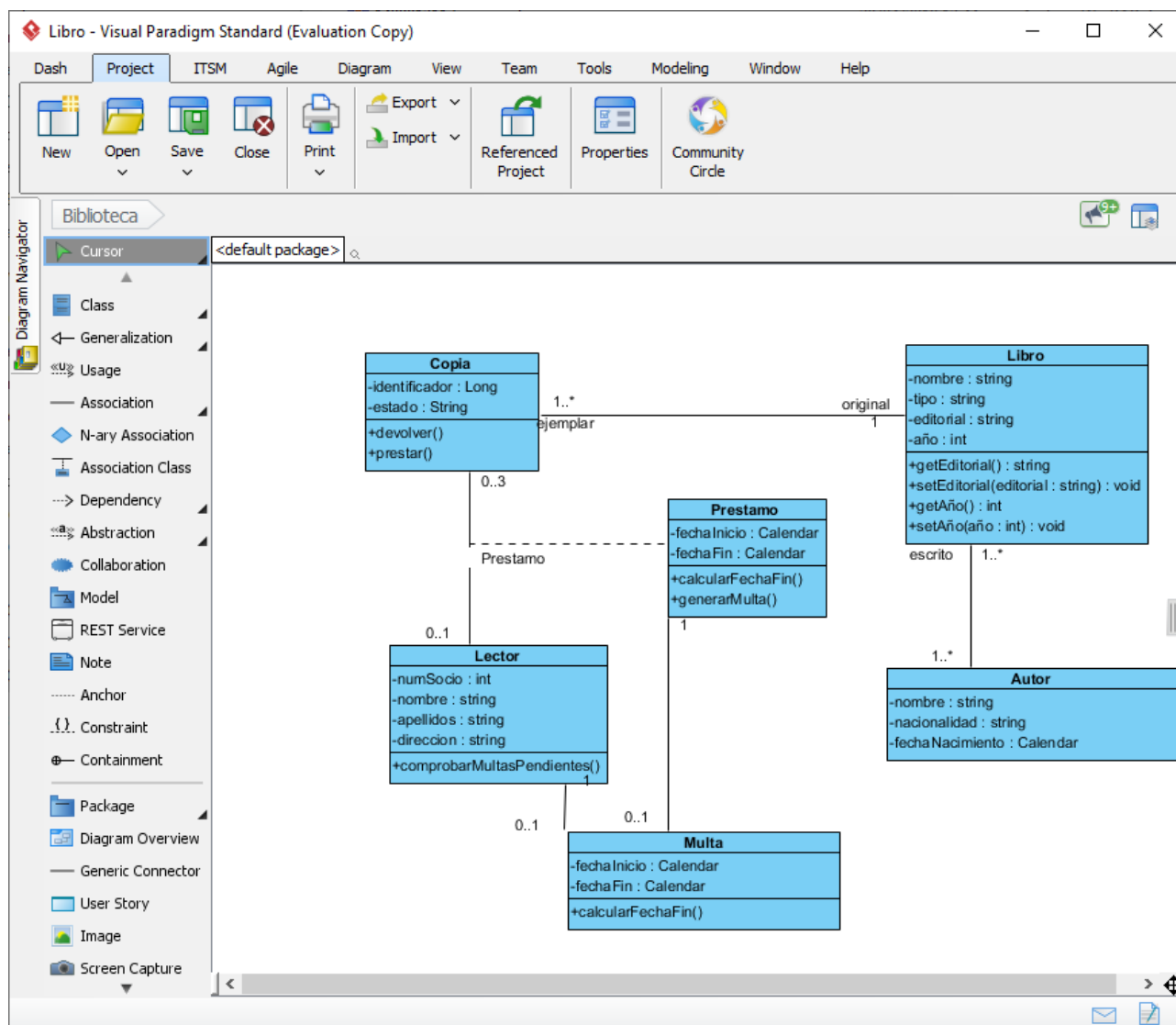
SSL Secure Connection

For Windows 64bit

[More Options](#) | [System Requirements](#) | [End User License Agreement](#)

Unha vez instalado, a primeira vez que o executamos pediranos que solicitemos unha clave de activación a través do enderezo web. Este paso non e necesario aínda que si recomendable xa que se non a aplicación estará recordándonos continuamente que traballamos cunha versión sen activar:

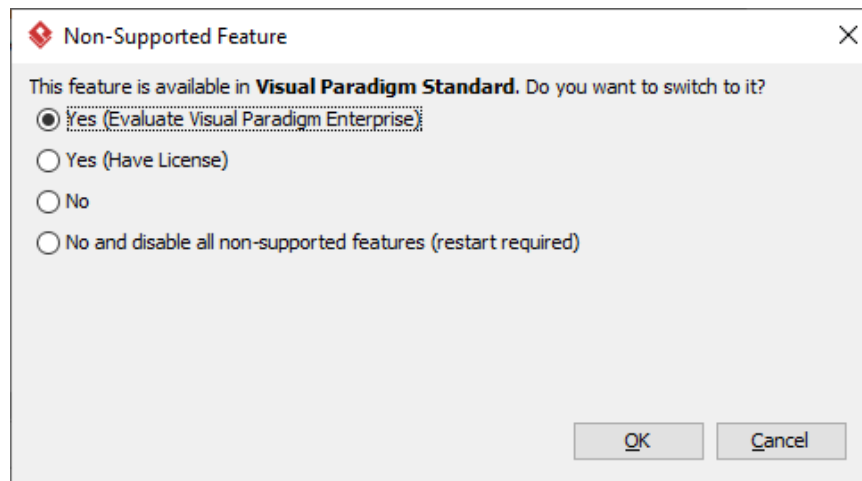
Finalmente podemos crear un novo proxecto e comezar a traballar:



### Xeración de código a partir de diagramas de clases

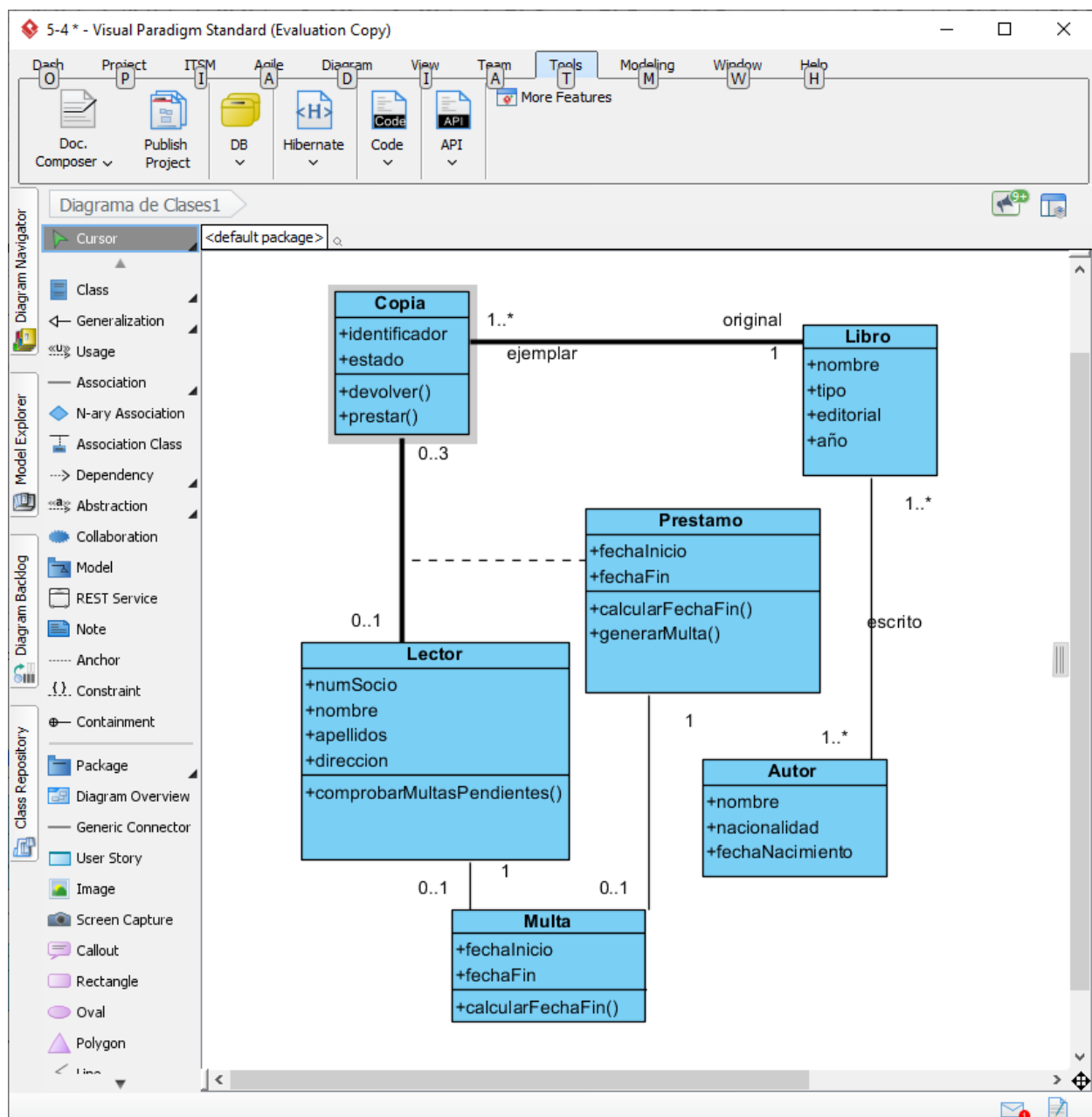
Tanto para a xeración de código como para o traballo con enxeñería inversa, Visual Paradigm permite traballar cunha tecnoloxía que chaman *Round-Trip* de tal forma que calquera modificación que fagamos no código das nosas clases ou no propio diagrama UML van verse reflectidos automaticamente nos diagramas UML ou no código fonte respectivamente. Esta funcionalidade permite manter ó código fonte e a documentación de deseño sincronizadas.

A versión gratuíta de Visual Paradigm non permite o traballo con estas funcionalidades de Enxeñería de Código (Submenú *Tools->Code->Reverse Java Code*) polo que teremos que utilizar a versión de proba de 30 días. Dende a propia versión *Community Edition* permítesenos facer o cambio (non é necesario descargar e instalar outra versión). Cando accedemos a unha funcionalidade non soportada pola versión gratuíta saíranos esta pantalla:

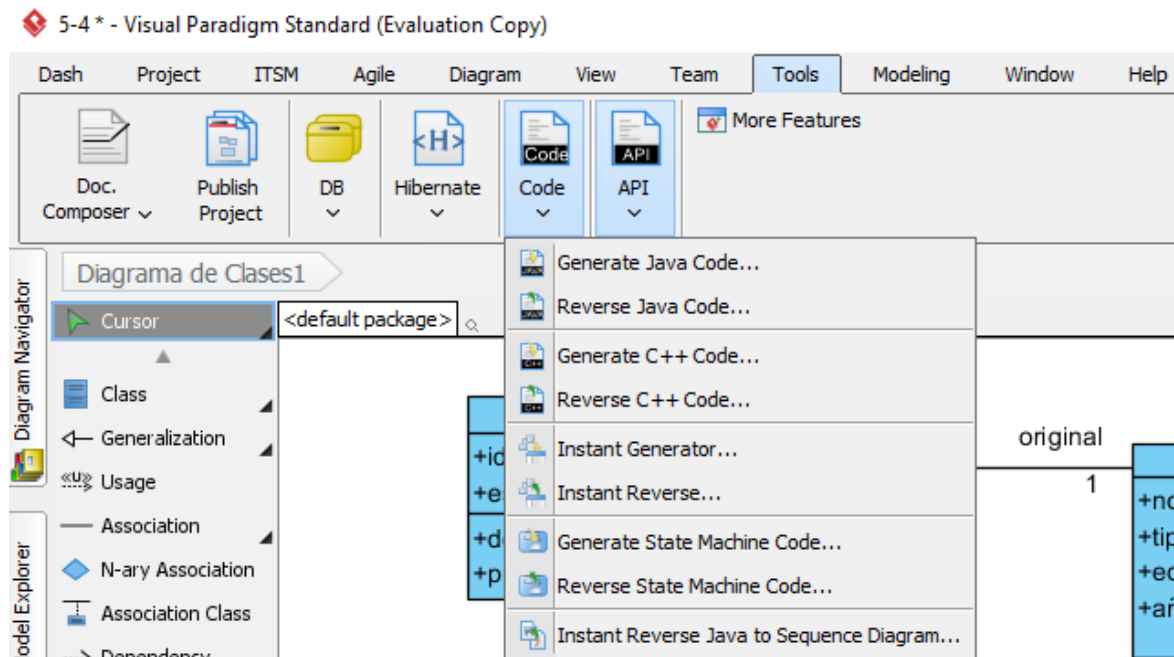


Se confirmamos o cambio guíaranos para empezar a utilizar a versión de proba (sempre poderemos volver a versión gratuíta cando queiramos).

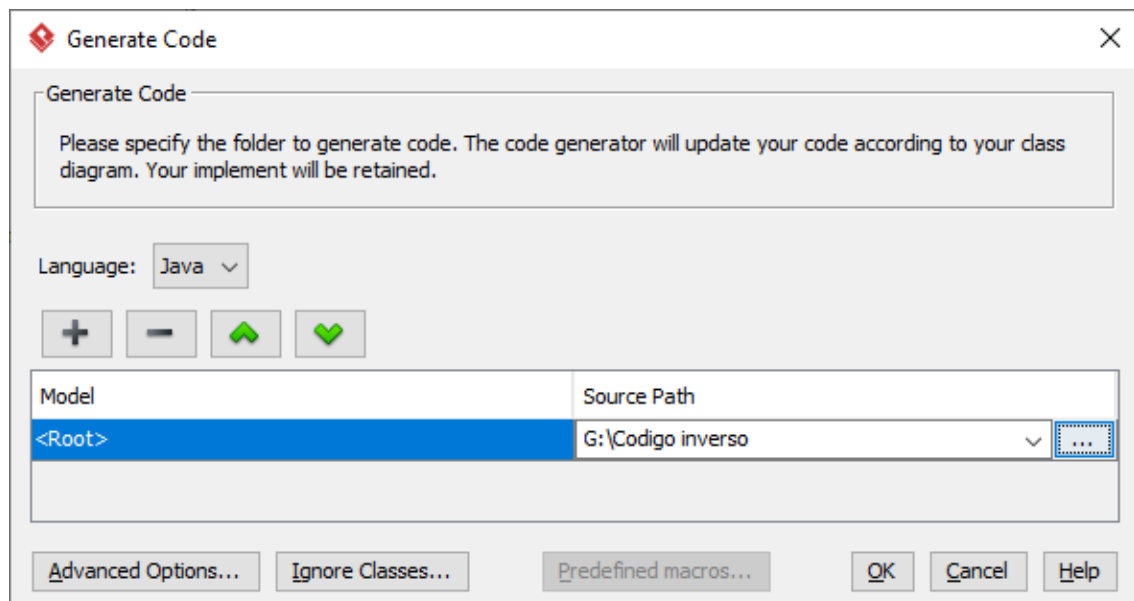
Unha vez na versión de proba, seleccionamos o proxecto que contén o diagrama de clases a partir do cal queremos xerar código:



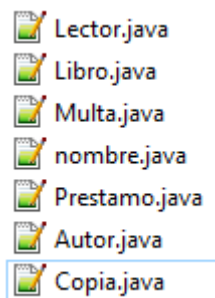
Imos a opción *Tools-> Code-> Generate Java Code*:



Seleccionamos o directorio destino onde queremos que se garden os ficheiros xerados e pulsamos *OK*. Se queremos personalizar o código xerado podemos modificar varios parámetros no botón *AdvancedOptions*.



Podemos ver no directorio os ficheiros creados:



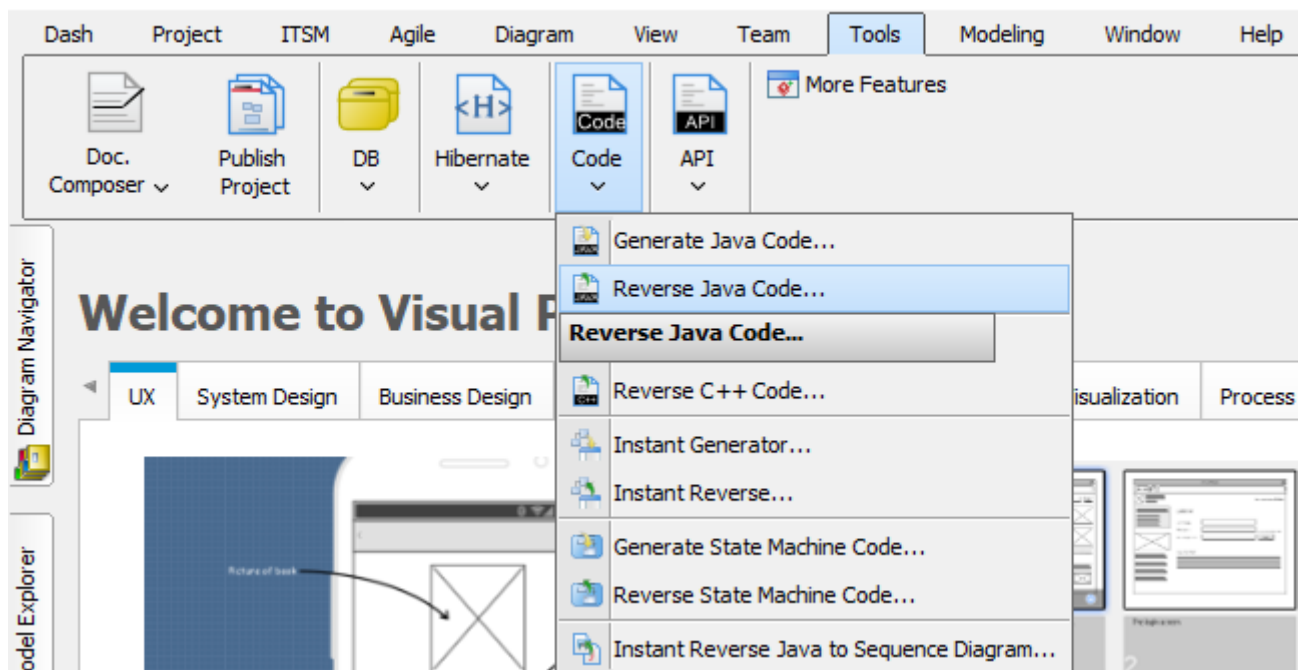
E o código dalgún dos ficheiros:

```
1 public class Lector {
2
3     public int numSocio;
4     public int nombre;
5     public int apellidos;
6     public int direccion;
7
8     public void comprobarMultasPendientes() {
9         throw new UnsupportedOperationException();
10    }
11
12 }
```

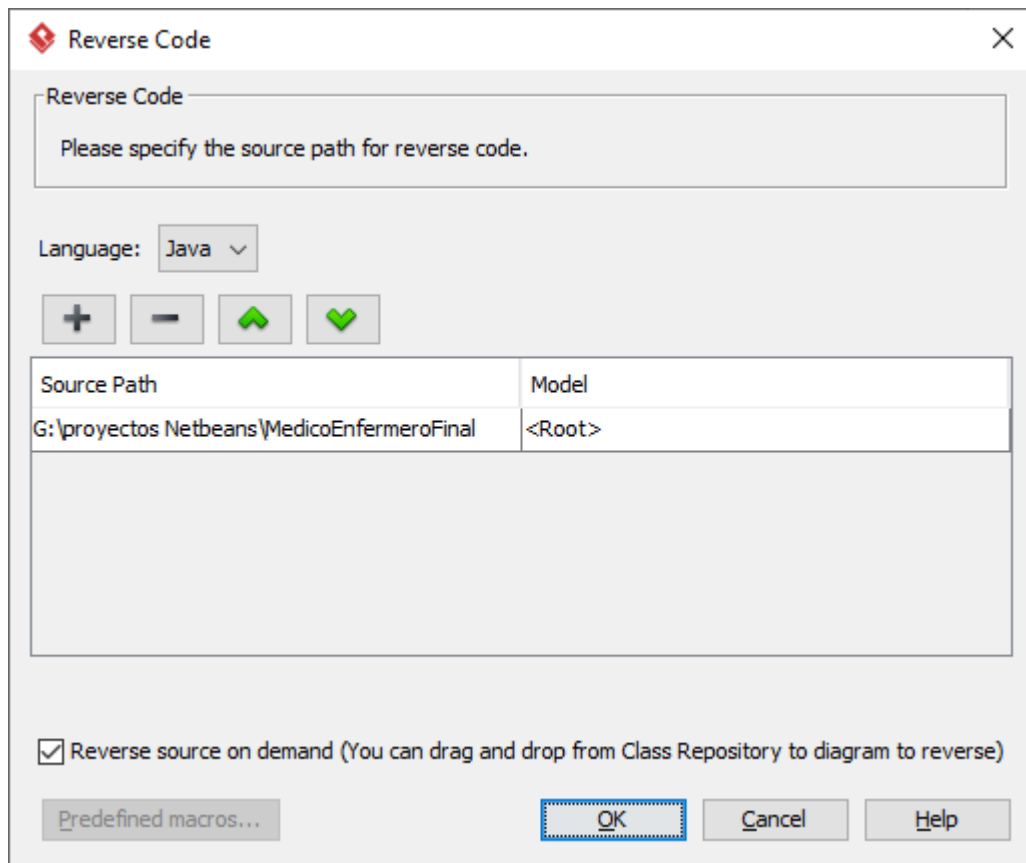
### Xeración de diagramas de clase a partir de código (enxeñaría inversa)

Para xerar diagramas de clase a partir de código fonte xa existente primeiramente creamos un proxecto novo no que imos crear os nosos diagramas e a continuación iremos ó submenú *Tools->Code ->Reverse Code* (neste caso crearemos os diagramas de clases do código Java do proxecto Médico-Enfermeiro):

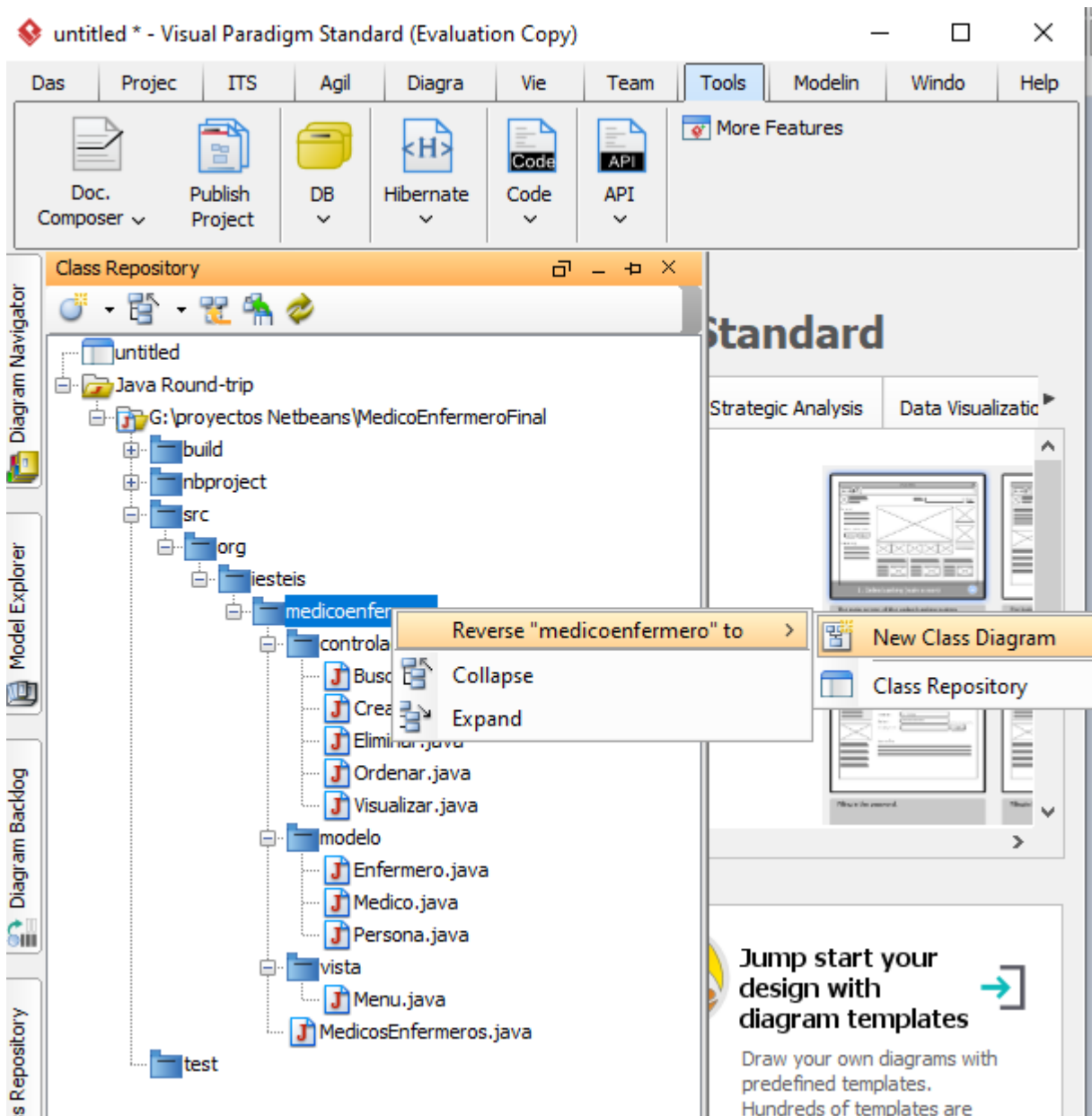
untitled - Visual Paradigm Standard (Evaluation Copy)



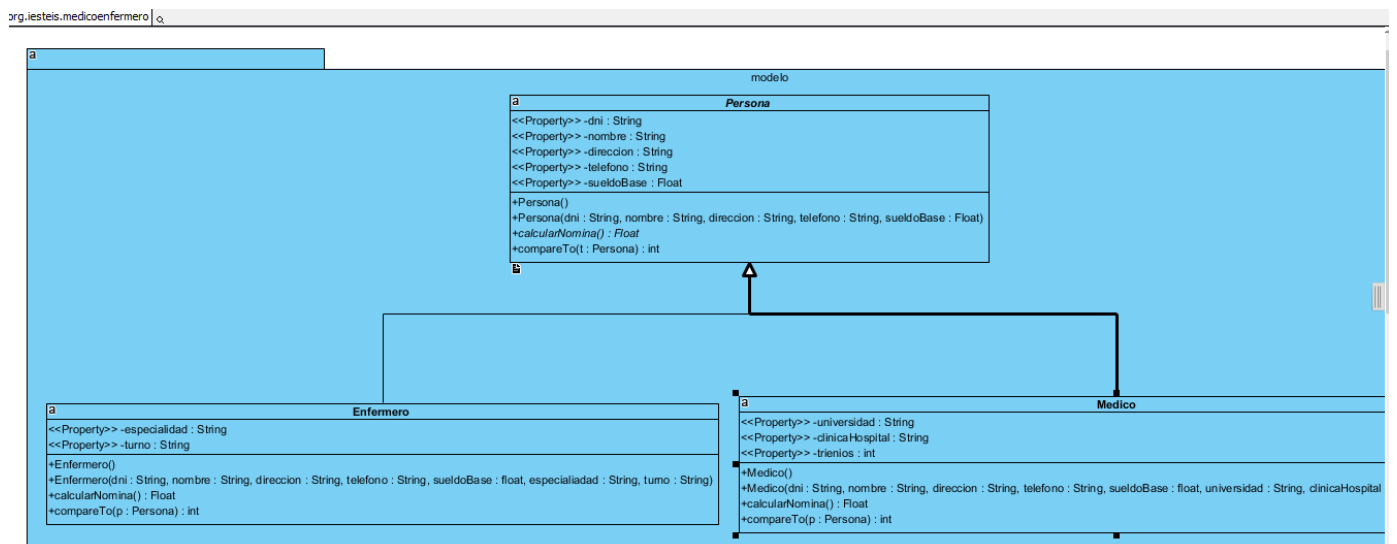
Seleccionamos o directorio onde están os ficheiros có código fonte e pulsamos *OK*:



Despois disto, no repositorio de clases do menú esquerdo aparecerannos as clases atopadas. Seleccionamos todas e có botón dereito do rato dicímoslle que xere un novo diagrama de clases:



Finalmente vemos o diagrama xerado:







Realizar a Tarefa 2 consistente en instalar Visual Paradigm, crear diagramas de clases, xerar código e facer enxeñería inversa.

## 1.3 Tarefas

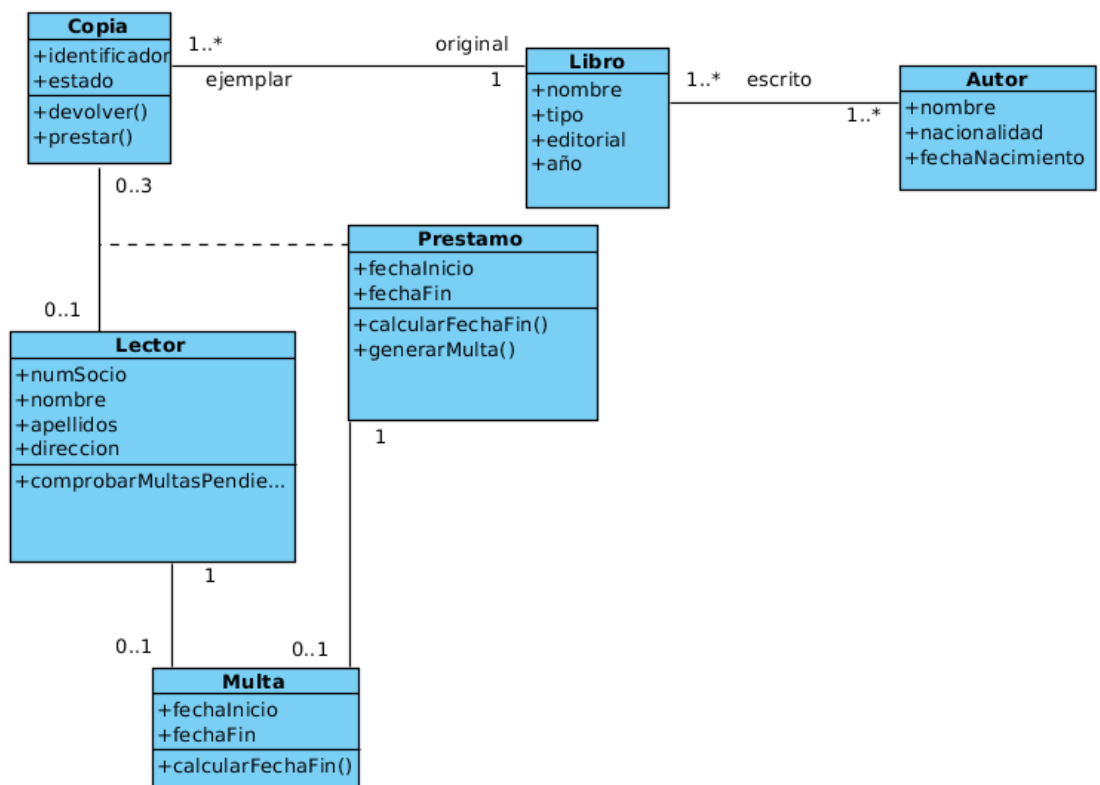
As tarefas propostas son as seguintes.

- **Tarefa 1.** *Instalación e manexo do módulo easyUML de NetBeans.* Nesta tarefa imos traballar co módulo easyUML de Netbeans representando un diagrama de clases e manexando as funcionalidades de xeración de código.
- **Tarefa 2.** *Instalación e manexo de Visual Paradigm.* Nesta tarefa repetiremos os apartados da tarefa anterior pero utilizando unha ferramenta non integrada nun contorno de desenvolvemento como é Visual Paradigm.

### 1.3.1 Tarefa 1. Instalación e manexo do módulo UML de NetBeans

Empregando o teu contorno de desenvolvemento NetBeans, realiza os seguintes apartados:

- Seguindo as indicacións dadas no texto da actividade instala o módulo de UML no contorno de desenvolvemento.
- Usando o módulo UML instalado representa o seguinte diagrama de clases:



Crea en NetBeans un novo proxecto Java e xera automaticamente código a partir do diagrama anterior.

### 1.3.2 Tarefa 2. Instalación e manexo de Visual Paradigm

Seguindo as indicacións dadas no texto da actividade, instala a versión gratuíta de Visual Paradigm e despois realiza os mesmos apartados da tarefa 1 pero esta vez xera código en linguaxe C++. Recorda que para a xeración automática de código e para a enxeñería inversa deberás de cambiar á versión de proba de Visual Paradigm.