



EPS

Escola Politècnica

Superior

Projecte/Treball Fi de Carrera

Estudi: Enginyeria Informàtica. Pla 1997

Títol: Desenvolupament d'una biblioteca per a la generació de diagrames vectorials basat en l'entorn gràfic de JAVA2D i aplicació d'aquesta biblioteca per a la implementació d'una eina per al desenvolupament de diagrames d'E/R

Document: Resum

Alumne: CRISTINA CONSTANTINS ESGLEYES

Director/Tutor: JOSEP SUY FRANCH

Departament: Informàtica i Matemàtica Aplicada

Àrea: Llenguatges i Sistemes Informàtics

Convocatòria (mes/any): setembre 2008

1.- INDEX

1.-	INDEX	1
2.-	INTRODUCCIÓ.....	2
2.1.-	OBJECTIUS	2
2.2.-	ABAST	2
3.-	ESTUDIS PRÈVIS	3
4.-	ESTUDI DE LA TECNOLOGIA SOFTWARE	3
5.-	ANÀLISI DE REQUERIMENTS	4
6.-	ANÀLISI I DISSENY.....	4
7.-	IMPLEMENTACIÓ.....	5
7.1.-	PERSISTÈNCIA I EXPORTACIÓ	6
8.-	EDITOR ENTITAT/RELACIÓ.....	6
9.-	CONCLUSIONS.....	7

2.- INTRODUCCIÓ

En aquest projecte es vol implementar una llibreria en Java, que proporcioni les eines necessàries per a crear diagrames i esquemes. Es vol proporcionar a la llibreria un conjunt d'objectes bàsics, també mètodes i funcionalitats suficients per a gestionar aquests objectes.

Per altra part, es vol implementar una aplicació per a dissenyar diagrames Entitat-Relació que utilitzi aquesta llibreria.

2.1.- OBJECTIUS

El principal objectiu d'aquest projecte és obtenir una eina de programació senzilla per a generar diagrames vectorials en dos dimensions. Aquesta eina ha de poder ser incorporada en diferents tipus de projectes o aplicacions que necessitin una interfície visual per interactuar amb l'usuari.

Per a representar els diferents diagrames, la llibreria ha de proporcionar un conjunt d'objectes bàsics, com també un conjunt de funcionalitats mínimes per a tractar aquests objectes.

La llibreria ha de proporcionar un mecanisme per a poder emmagatzemar els objectes que s'instancien i les propietats de manera que es puguin recuperar posteriorment.

Finalment es desenvoluparà un editor per a generar diagrames Entitat-Relació que utilitzi la llibreria dissenyada. Mitjançant aquesta aplicació es mostrarà una de les diferents formes de com pot, una aplicació, utilitzar la llibreria com la seva interfície d'usuari. Aquest editor ha de permetre crear els objectes principals d'una base de dades i realitzar diferents funcionalitats en aquest camp, com per exemple obtenir un informe de les taules i relacions instanciades o obtenir un script per a crear les taules en una base de dades.

2.2.- ABAST

Al realitzar un projecte una mica gran i complet és necessari seguir una metodologia de treball, per a aquest projecte s'ha escollit eXtreme Programming, ja que permet anar implementant el codi a mesura que es van acabant de dissenyar les diferents parts.

En el desenvolupament del projecte no es disposa d'un client extern, de manera que les diferents iteracions s'aniran provant i avaluant com a part del mateix projecte.

S'ha decidit utilitzar UML 1 (Unified Modeling Language) com a llenguatge de suport per a modelar i documentar el projecte principalment perquè està molt enfocat al desenvolupament de projectes Orientats a Objectes.

Per a desenvolupar el projecte s'ha buscat el suport d'una eina de desenvolupament. Per a Java les eines de desenvolupament que destaquen més són NetBeans i Eclipse. M'he decantat per utilitzar Netbeans perquè és una eina amb la que ja havia fet alguna petita aplicació, és bastant senzilla d'instal·lar i d'utilitzar, i a més, està disponible per a diverses plataformes (Windows, Linux, Unix, etc.).

3 .- ESTUDIS PRÈVIS

En el mercat ja existeixen llibreries gràfiques que permeten construir dibuixos o diagrames. Abans de definir els requeriments s'ha realitzat un petit anàlisi d'aquestes eines.

S'han avaluat llibreries com la JGraph, GEF, OpenJGraph, yFiles o CADE Control, la majoria d'elles compatibles amb Java. S'ha observat que cada llibreria implementa un conjunt d'operacions i funcionalitats diferents.

Per altra part s'ha realitzat un estudi similar respecte algunes aplicacions disponibles en el mercat per a realitzar diagrames Entitat-Relació. S'ha analitzat el Microsoft Access, el DBDesigner, el Dia i l'Oracle Designer. La principal característica que s'ha pogut comprovar que són molt poques les que permeten generar scripts, altres actuen directament sobre les bases de dades de la casa comercial que les desenvolupa.

4 .- ESTUDI DE LA TECNOLOGIA SOFTWARE

En aquest punt es descriu la tecnologia software i les tècniques de disseny que s'ha utilitzat per al desenvolupament d'aquest projecte.

Com a llenguatge de programació s'ha escollit el Java perquè es tracta d'un llenguatge de programació orientat a objectes i per la seva capacitat multiplataforma. Per al desenvolupament del codi s'ha utilitzat una IDE (Netbeans). Netbeans disposa d'un editor de codi sensible al contingut i suport per autocompletar el codi i permet la creació visual de component gràfics.

S'ha utilitzat l'UML com a suport per l'anàlisi i el disseny del projecte. Per a realitzar els diagrames s'ha utilitzat el programa UML Diagrammer de l'empresa Pacestar Software.

La llibreria utilitza el llenguatge d'etiquetes XML per a guardar els projectes. S'ha utilitzat l'ANTLR per a realitzar el parser per reconèixer l'XML.

5 .- ANÀLISI DE REQUERIMENTS

Al tractar-se d'un entorn visual és necessari un conjunt bastant ampli de requeriments per a poder definir les funcionalitats bàsiques per a la llibreria. Al analitzar els requeriments s'ha tenint en compte els que afecten l'entorn i els que afecten els elements.

5.1.1 .- Entorn

S'ha considerat un conjunt de requeriments per a aconseguir proporcionar un entorn de treball simple i fàcil de manejar.

Entre els requeriments de l'entorn de treball cal destacar la definició d'una o més vistes on es representen els elements i recullen els events que permeten la realització de les operacions de la llibreria. També destacar les operacions que es poden realitzar amb els elements, com per exemple seleccionar-los, moure'ls per la vista, copiar-los i enganxar-los posteriorment. La llibreria permet treballar utilitzant capes, de manera que en crear-se un element pren les propietats de la capa on es crea.

5.1.2 .- Objectes

La llibreria proporciona un conjunt d'elements bàsics com línies, rectangles, ... i altres una mica no tan bàsics com polilínies o connectors. També s'ha inclòs la funcionalitat de poder agrupar figures i crear-ne de més complexes. Cada element gestiona les seves propietats de color, dimensions, forma, ...

Les diferents figures es poden bloquejar de manera no es puguin modificar les seves propietats ni rebre events com per exemple el moure, també es poden fer invisibles.

Paral·lelament a la llibreria s'ha realitzat una petita aplicació per anar realitzant les diferents proves.

6 .- ANÀLISI I DISSENY

En la part d'anàlisi i disseny es presenta l'estructura de la llibreria i se'n defineixen les classes. Al dissenyar la llibreria s'ha observat la necessitat d'aplicar alguns patrons. Amb aquests patrons s'aconsegueix simplificar el problema.

La llibreria es basa en el patró arquitectònic MVC per a separar el que és la visualització del que són els objectes i les classes que fan de control.

El model està format per el conjunt d'elements bàsics per a realitzar diagrames. Per a representar els diferents tipus d'elements s'ha utilitzat una jerarquia de classes que

comença en l'interfície ElementDibuixable. El sistema de capes també forma part del model.

La vista de la llibreria està formada per les classes vista, on es mostren els elements de la llibreria.

L'objecte mediador és l'objecte de control més destacat de la llibreria, ja que és l'encarregat de gestionar la majoria de les operacions que permet realitzar la llibreria. El mecanisme d'estats també forma part del control de la llibreria per què és qui controla les operacions que cal realitzar quan l'usuari provoca un event.

7.- IMPLEMENTACIÓ

En la implementació s'ha decidit mantenir algunes convencions ja provinents del disseny en un entorn gràfic 2D.

La vista és la part central de la llibreria, ja que és sobre ella que es pinten els diferents objectes, s'ha implementat la classe VistaEstandar que hereta de la classe JPanel de Java. La classe JPanel de Java proporciona la interfície necessària per a poder representar els diferents elements de la llibreria. Per a proporcionar una sensació de manipulació directa dels elements s'implementa les interfícies KeyListener, MouseListener, MouseMotionListener i MouseWheelListener de Java.

S'ha implementat diferents classes per a representen el concepte punt, cada classe amb propietats i funcionalitats diferents. Per a guardar la posició, les dimensions i la forma dels elements s'utilitzen conjunts de punts.

Els elements permeten que es realitzin un ampli ventall d'operacions sobre ells. Aquestes operacions sovint necessiten realitzar un conjunt de passos per a dur-se a terme, aquests passos s'han implementat en forma d'estats.

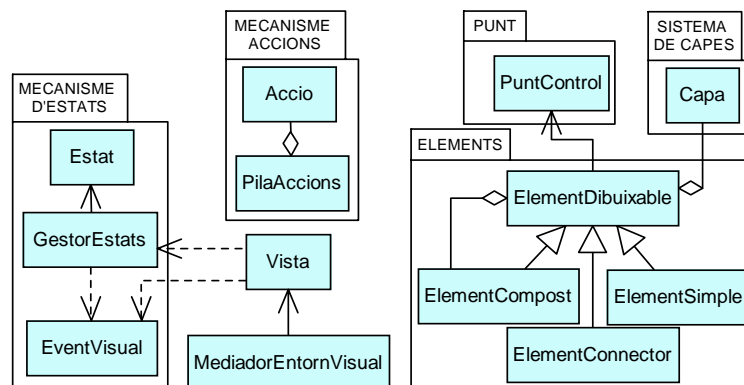


Figura 1.- Esquema de les principals classes de la llibreria

7.1 .- PERSISTÈNCIA I EXPORTACIÓ

En la llibreria s'ha utilitzat dos mecanismes per a guardar l'entorn i els objectes: la serialització de Java i la traducció a una particularització d'XML. Per a reconèixer els objectes guardats en l'XML, s'ha creat un petit parser en ANTLR.

La llibreria també implementa un mecanisme per a exportar a imatge, definint la part de la imatge a exportar mitjançant dos punts. Per definir els dos punts es pot fer amb el ratolí o bé utilitzant el mètode del Mediator.

8 .- EDITOR ENTITAT/RELACIÓ

Abans d'analitzar els requeriments s'ha realitzat un recull de conceptes de bases de dades sobre la definició de models Entitat-Relació i el model relacional.

En l'anàlisi de requeriments de l'aplicació s'ha tingut en compte sols aquells requeriments que tenen a veure exclusivament amb l'aplicació: gestionar projectes, imprimir, check, exportar a script,

L'anàlisi de classes s'ha fet pensant ja en una estructura MVC, on un conjunt de classes s'encarreguen de tot el que és la visualització de l'entorn i dels objectes, en aquest conjunt s'hi inclou la llibreria implementada en el projecte, un altre conjunt de classes representa el model d'objectes de l'aplicació i per últim unes altres classes fan de controladores per a mantenir la coherència i la integritat entre el model i la visualització.

L'aplicació consta d'una pantalla principal i un conjunt de diàlegs que permeten realitzar operacions secundàries com: modificar les propietats dels elements, les propietats generals de l'aplicació, S'ha realitzat un prototipus per a definir la pantalla principal de l'aplicació.

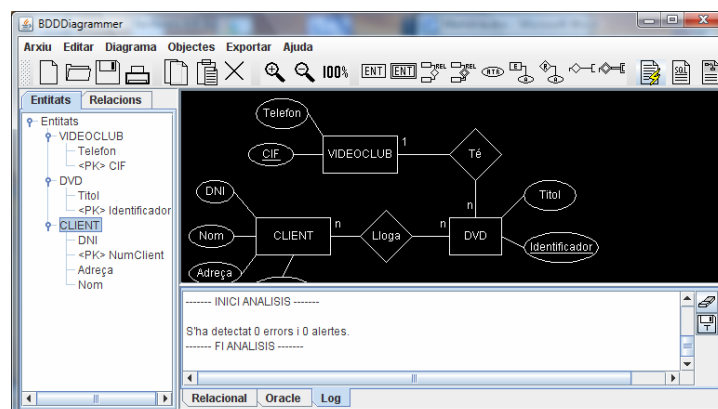


Figura 2.- Pantalla principal de l'aplicació BDDDiagrammer

L'aplicació interacciona amb la llibreria principalment utilitzant l'objecte Mediator de la llibreria. Un altre punt d'interacció és a través de la redefinició de la vista estàndard que proporciona la llibreria.

Una altra part destacable de l'aplicació és l'anàlisi i exportació a script o HTML. L'anàlisi s'encarrega d'avaluar que no hi hagi incoherències en el diagrama dissenyat i en cas de superar-se positivament es permet exportar a script per a bases de dades Oracle. S'ha comprovat els scripts generats amb l'aplicació en una base de dades Oracle 10g. Per a exportar a HTML es transcriu automàticament el diagrama dissenyat al model relacional.

9.- CONCLUSIONS

El present projecte tracta el disseny d'una llibreria per a facilitar el disseny d'aplicacions que utilitzen diagrames com a interfície d'usuari i l'aplicació d'aquesta llibreria en un editor per a dissenyar diagrames Entitat-Relació. Es considera que els objectius del projecte s'han pogut assolir satisfactòriament.

A l'inici del projecte es va plantejar una planificació. Al final del projecte s'ha pogut comprovar que la temporalització s'ha seguit en gran mesura, sols amb petites modificacions degudes sobretot a obligacions externes al projecte.

En la fase d'anàlisi del projecte s'han comentat els problemes i les solucions escollides amb Artur Juvé, un altre projectista que es troba desenvolupant un projecte similar. El fet de comentar el disseny de les diferents parts amb l'Artur ha estat decisiu per a dur a terme algunes funcionalitats que en un primer moment s'havien descartat. També ha estat molt productiu ja que d'aquesta manera s'han observat moltes més casuístiques i solucions que si no s'hagués disposat d'algú amb qui poder-ho comentar.

En l'anàlisi de requeriments s'han avaluat un ampli ventall de funcionalitats i s'ha definit un conjunt bàsic a implementar per a obtenir una llibreria que permeti un tractament simple dels elements. El mateix s'ha fet amb l'aplicació BDDDDiagrammer.

Com a treball futur es preveu ampliar la llibreria implementant noves funcionalitats que no s'ha considerat bàsiques i millorant les funcionalitat de les que s'ha realitzat una implementació bàsica.