

U . M . L . (UNIFIED MODELING LANGUAGE)

Concetto Base :

- UML permette, tramite l'utilizzo di modelli visuali, di analizzare, descrivere, specificare e documentare un sistema software anche complesso. UML consente di avere una visione chiara del sistema anche alle persone con figure professionali molto diverse tra loro.

Esigenze dello sviluppo :

- Nel mondo costantemente in fermento ed in evoluzione quale e' quello dello sviluppo di applicazioni Object Oriented, diviene sempre più difficile costruire e gestire applicazioni di alta qualità in tempi brevi.

Come risultato di tale difficoltà e dalla esigenza di avere un linguaggio universale per modellare gli oggetti che potesse essere utilizzato da ogni industria produttrice di software, fu inventato il linguaggio **UML**.

Preparazione del modello :

- In termini pratici, si può dire che il linguaggio UML rappresenta, in qualche modo, un progetto di ingegneria edile, riportato nel mondo dell'Information Technology.

L'UML è, dunque, un metodo per descrivere l'architettura di un sistema in dettaglio. Come è facile intuire, utilizzando un processo grafico sarà molto più facile costruire o mantenere un sistema e assicurarsi che il sistema stesso si presterà senza troppe difficoltà a futuri cambiamenti dei requisiti.

- La forza dell' U.M.L. consiste nel fatto che il processo di disegno del sistema può essere effettuata in modo tale che i clienti, gli analisti, i programmatori e chiunque altro sia coinvolto nel sistema di sviluppo possa capire ed esaminare in modo efficiente il sistema e prendere parte alla sua costruzione in modo attivo.

I diagrammi UML non fanno altro che documentare in modo esauriente e a tutti i livelli le caratteristiche tecniche che dovranno essere implementate.

Alcuni dei vantaggi derivanti dall'utilizzo del linguaggio UML:

1. Un sistema software grazie al linguaggio UML viene **disegnato professionalmente e documentato** ancor prima che ne venga scritto il relativo codice, da parte degli sviluppatori. Si sarà così in grado di conoscere in anticipo il risultato finale del progetto su cui si sta lavorando.
2. Poichè, la fase di disegno del sistema precede la fase di scrittura del codice, ne consegue che la **scrittura del codice stessa è resa più agevole ed efficiente** oltre al fatto che in tal modo è più facile scrivere del **codice riutilizzabile in futuro**. I costi di sviluppo, dunque, si abbassano notevolmente con l'utilizzo del linguaggio UML.
3. È più facile **prevedere e anticipare eventuali "buchi"** nel sistema. Il software che si scrive, si comporterà esattamente come ci si aspetta senza spiacevoli sorprese finali.
4. L'utilizzo dei diagrammi UML permette di avere una **chiara idea**, a chiunque sia coinvolto nello sviluppo, di tutto l'insieme che costituisce il sistema. In questo modo, si potranno sfruttare al meglio anche le risorse hardware in termini di memoria ed efficienza, senza sprechi inutili o, al contrario, rischi di sottostima dei requisiti di sistema.
5. Grazie alla documentazione del linguaggio UML diviene ancora più **facile effettuare eventuali modifiche future** al codice.
6. La comunicazione e l'interazione **tra tutte le risorse umane** che prendono parte allo sviluppo del sistema è molto più efficiente e diretta. Si evitano rischi di incomprensioni e quindi sprechi di tempo.

U . M . L . - TOOLS DI SVILUPPO -

Il linguaggio UML contiene svariati elementi grafici che vengono messi insieme durante la creazione dei diagrammi. Poiché l'UML è un linguaggio, esso utilizza delle regole per combinare i componenti del linguaggio nella creazione dei diagrammi. L'obiettivo dei diagrammi è quello di costruire molteplici viste di un sistema tutte correlate tra di loro. L'insieme di tali viste costituirà il **Visual Modeling**.

Come ogni linguaggio che si rispetti anche l'UML necessita dell'utilizzo di tools appropriati che ne agevolino l'utilizzo.

La scelta di un Tool un UML è un passo importante quando si inizia un processo di sviluppo basato sull'UML. Infatti tutti i tool basati su UML implementano una particolare metodologia di sviluppo, alcuni permettono di scegliere fra diverse possibilità. In generale si dovrebbe comunque scegliere la metodologia più adatta al tipo di progetto che si deve affrontare e di conseguenza scegliere un tool che supporti o si adatti bene a tale metodologia

Di seguito i principali software per la produzione di diagrammi UML :

7. Il **Rational Rose** è stato disegnato per fornire ai team di sviluppo tutti gli strumenti necessari per il modellamento di soluzioni robuste ed efficienti. Rational Rose supporta la metodologia ER (Entity-Relationship).
 8. La Microsoft ha prodotto uno strumento che permette di definire un sottoinsieme dei modelli che il Rational Rose mette a disposizione: il **Microsoft Visual Modeler**. Diciamo che tale software può essere consigliato a chi si avvicina per la prima volta al mondo del Visual Modeling. Il Microsoft Visual Modeler permette, tra l'altro, di:
 1. Identificare e disegnare oggetti del business e mapparli in componenti software
 2. Descrivere come i componenti possono essere distribuiti su una rete
 3. Generare codice base Visual Basic direttamente dalla costruzione del modello
 4. Utilizzare il reverse engineering per creare i modelli da applicazioni già esistenti
- **UMLet** è un software open source basato su Java che permette la scrittura di diagrammi UML. Gli elementi UML sono modificabili tramite input testuali al posto dei tradizionali pop-up, inoltre questi possono essere usati come templates. UMLet supporta una varietà di diagrammi : di classe, use cases , di sequenza, di stato, di sviluppo, di attività etc.. Inoltre permette agli utenti di creare i propri elementi UML. La grafica di un elemento può essere modificata al run-time cambiando solo alcune linee di codice Java; UMLet compila immediatamente il codice del nuovo elemento. Senza uscire da UMLet, gli utenti possono creare e aggiungere nuovi tipi di elementi ai propri diagrammi. Tra le opzioni, la possibilità di esportare i diagrammi in SVG, JPG e PDF.
 - **Umbrello UML Modeler** è un programma OpenSource per la modellazione visuale del software con l'Unified Modelling Language (UML).
Tra le caratteristiche più importanti di Umbrello c'è quella di poter generare il codice sorgente in più di 10 linguaggi diversi, tra i quali: C++, Java, PHP, SQL e Python.
E' scritto in C++ per l'ambiente KDE ed è rilasciato sotto licenza GPL.
 - **ArgoUML** è un software grafico user-friendly per il disegno, lo sviluppo e la documentazione di applicazioni orientate agli oggetti. Gli utenti che utilizzano ArgoUml sono designers, programmatori, analisti di business, analisti di sistema ed altre figure professionali coinvolti nell' analisi, il disegno e lo sviluppo di software. Le principali caratteristiche sono :
 1. Compatibilità con XMI, SVG and PGML
 2. Indipendenza completa dal tipo di piattaforma utilizzata grazie all' esclusivo uso di codice Java
 3. Open Source