

Appalto concorso per la realizzazione di un sistema *software* per la generazione automatica di applicazioni *software* in ambito gestionale (GAAS)

Art. 1) Oggetto dell'appalto

Il presente capitolato concerne l'affidamento a fornitori terzi, della realizzazione di un sistema *software* prototipale la generazione automatica di applicazioni *software* in ambito gestionale.

Il termine "**GAAS**" denota il prodotto *software* in oggetto.

Il termine "committente" denota i proff. Vardanega e Conte, in qualità di delegati rappresentanti dell'azienda che ha commissionato il lavoro.

Art. 2) Caratteristiche e requisiti tecnici minimi

Premessa

Il progetto qui illustrato è intitolato:

Automazione dello sviluppo di sistemi informatici gestionali in architettura J2EE

Tale progetto, nel seguito denominato "progetto specifico", si colloca nel quadro di un progetto aziendale strategico più ampio, nel seguito denominato "progetto globale".

Obiettivo del progetto globale

Il progetto specifico fa parte di un più ampio progetto globale intrapreso dall'azienda committente, i cui obiettivi sono i seguenti:

- ✚ automazione nello sviluppo di sistemi informatici "gestionali" transazionali basati su *database* relazionali
- ✚ definizione delle specifiche applicative attraverso la definizione di un apposito sistema di regole espresse in modo dichiarativo e/o tramite *script* controllabili e manipolabili dall'utente
- ✚ generazione automatica del codice applicativo a partire dalle regole di cui sopra mediante l'utilizzo di opportuni modelli¹
- ✚ utilizzo di modelli di generazione *multi-language*, capaci di generare codice sorgente in svariati linguaggi di programmazione e di aderire eventuali altre caratteristiche desiderate.

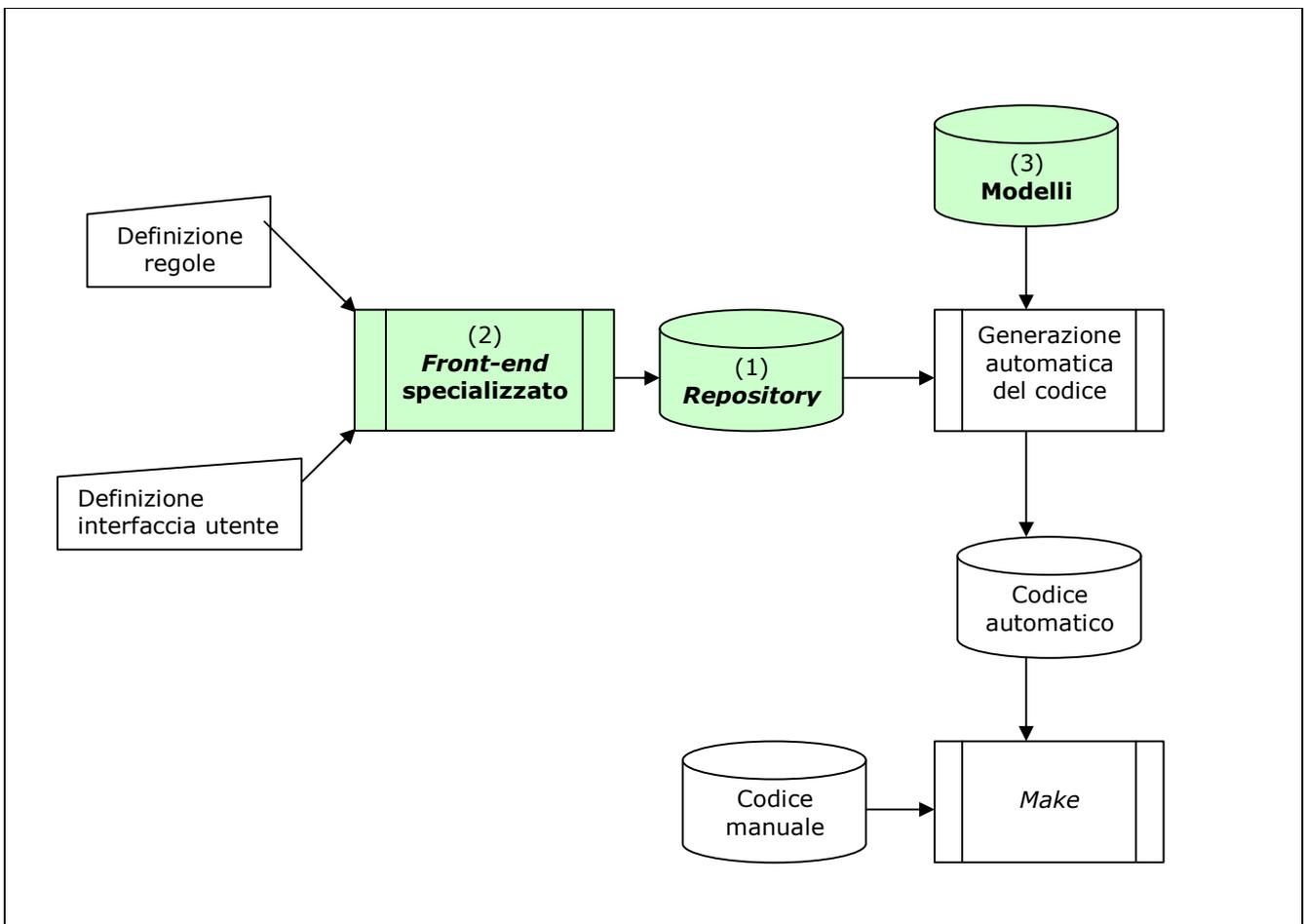
Un sistema dotato di tali caratteristiche otterrebbe i seguenti immediati benefici:

- ✚ maggiore **produttività** (il codice viene generato automaticamente ed una regola contenuta in una sola riga di testo sostituisce decine o centinaia di righe di codice)

¹ I modelli sono porzioni di codice sorgente del linguaggio *target*, gestiti con un linguaggio proprietario di lettura ed interpretazione delle regole definite nel *repository*.

- ✚ maggiore **qualità** (ancora: il codice viene generato automaticamente, esonerando lo sviluppatore dal ricordare quando una certa regola di programmazione deve scattare ed in quale sequenza all'interno di una particolare transazione)
- ✚ migliore **manutenzione** applicativa (poiché essa si sposta al livello delle regole e non più del codice scritto manualmente, e quindi per definizione auto-documentante e slegato dallo sviluppatore)
- ✚ immediato **adeguamento** alle novità tecnologiche (l'aggiornamento periodico dei modelli consente la rigenerazione totale del codice con nuove caratteristiche applicative (p.es. gestione accessi) o tecniche (p.es. interfaccia verso un nuovo DBMS).

Schema a blocchi del sistema complessivo



Stato dell'arte

Allo stato attuale, l'azienda committente ha realizzato in proprio i seguenti blocchi del sistema mostrato in figura:

1. *repository* delle regole (basato su XML)
2. *front-end* per l'immissione delle regole e della interfaccia utente
3. modelli di generazione in architettura *client-server* per codice Informix-4GL e VisualBasic.NET.

Il fornitore vincitore dell'appalto avrà a disposizione il prodotto alla data (*repository*, *front-end*, modelli realizzati), corredato da tutta la documentazione disponibile ed avrà facoltà di libera installazione di esso sulle piattaforme da esso ritenute opportune.

L'azienda committente è anche disponibile ad effettuare un corso iniziale diretto di avviamento della durata massima di due giornate lavorative.

Obiettivo del progetto specifico

Partendo dagli obiettivi del progetto globale come precedentemente illustrati ed alla luce dello stato dell'arte attuale, l'obiettivo primario dello specifico progetto è il seguente:

- ✚ generazione automatica del codice in ambiente *3-tier*, con utilizzo lato *server* dell'architettura **J2EE** ed utilizzo lato *client* di un normale **browser**.

Le singole macro-fasi del progetto dovranno essere le seguenti:

- ✚ raffronto e scelta dell'architettura migliore tra le varie possibili, sulla base dei seguenti criteri di valutazione:
 - adozione di chiari standard di mercato, affermati o in via di affermazione, p.es. *Apache*, *Struts* ed altri componenti
 - preferenza sistematica, ove possibile, per l'utilizzo di codice già esistente e preferibilmente *open-source*
 - assenza o minimizzazione di limitazioni teoriche determinate dalla scelta di componenti particolari come *application server*, *web-service* ed altro, in ogni caso sempre privilegiando la scelta di prodotti *open-source*.
- ✚ sviluppo di un prototipo iniziale, volto a verificare la fattibilità del progetto stesso con obiettivi specifici che verranno dettagliati nel corso dello sviluppo ma che verteranno essenzialmente su:
 - interazione tra i diversi livelli dell'architettura
 - realizzazione e verifica delle regole base
 - realizzazione e verifica degli oggetti base dell'interfaccia utente
- ✚ realizzazione del prodotto finale tramite:
 - sviluppo del prodotto
 - verifica e validazione
 - documentazione.

Per l'intera durata del progetto il fornitore potrà avere accesso ad una linea di posta elettronica diretta ad un referente designato dall'azienda committente, a disposizione per

tutte le necessità, mentre sarà eventualmente valutato nel merito, caso per caso, il tipo di intervento.

Sarà poi possibile prevedere riunioni tecniche di avanzamento tra il fornitore e l'azienda committente, con cadenza almeno quindicinale per la verifica dello stato avanzamento lavori e per l'assunzione di decisioni fondamentali per il prosieguo del progetto (in particolare nella fase di prototipo iniziale).

Definizione di ambiente J2EE

J2EE è un acronimo per **Java 2 Platform Enterprise Edition**.

J2EE è un ambiente platform-independent (indipendente dalla piattaforma hardware e software), Java-centric (basata sul linguaggio Java e su strumenti su questo basati), sviluppato da Sun Microsystems.

J2EE è nato per sviluppare e rilasciare applicazioni online basate sul Web.

La piattaforma **J2EE** consiste di una serie di servizi, API, protocolli che permettono di sviluppare e rilasciare applicazioni multitiered (multistrato, basati su livelli applicativi logici ben separati):

- livello client: strato interfaccia utente
- livello logic-server: strato di gestione della logica applicativa
- livello data-server: strato di gestione dei dati (DBMS)

A livello client J2EE supporta sia HTML puro sia *applets* Java. Si può basare su JSP (Java Server Pages) o su altre tecnologie di lato *server* per la preparazione dinamica delle pagine HTML.

A livello logic-server J2EE prevede i cosiddetti Enterprise JavaBeans (EJBs), tecnologia che fornisce accesso trasparente a servizi di *multithreading* e concorrenza, sicurezza e gestione della memoria.

A livello data-server J2EE prevede l'interfaccia Java Database Connectivity (JDBC), un'interfaccia standard equivalente allo standard ODBC del mondo MS Windows per la connessione e la gestione dei più popolari RDBMS.

Art. 3) La documentazione

Con la consegna del sistema **GAAS**, il fornitore assumerà anche l'onere di fornire i manuali ed ogni altra documentazione tecnica idonea per assicurare il soddisfacente funzionamento e manutenzione del prodotto, compresi i sorgenti del codice commentato secondo le norme.

La documentazione, in particolare, comprende i manuali e le istruzioni stabilite dall'impresa fornitrice, con i dettagli concernenti le caratteristiche dei programmi di base e le procedure per il loro utilizzo (avviamento, fermi, interventi per guasti, operazioni consentite in fase di elaborazione, ecc.).

Art. 5) Rinvio

Per tutto quanto non previsto nel presente capitolato, sono applicabili le vigenti disposizioni di legge.