

RMcube E-Business Java2EE Suite

White Paper



Finalità del documento

Lo scopo di questo documento è quello di fornire informazioni tecniche sulla filosofia, architettura ed implementazione del prodotto RMcube di Antica Bottega Digitale, società del gruppo Datamat, ponendo particolare riguardo a quegli aspetti funzionali che più caratterizzano il prodotto e la sua collocazione sul mercato.

Introduzione

RMcube nasce dallo stesso team di sviluppo che ha prodotto RoboMerchant V2, il primo software di commercio elettronico pensato in forma “pacchettizzata” sviluppato in Italia. RMcube è una suite per E-business sviluppata completamente in Java all’interno della piattaforma di riferimento J2EE definita da Sun Microsystems. Opera su molteplici piattaforme hardware e sistemi operativi, può impiegare tutti i principali Database Server presenti sul mercato. Fornisce un ambiente robusto e scalabile per la implementazione di applicazioni “web based”.

Filosofia del prodotto

Principi di sviluppo dell’applicazione RMcube

Il progetto RMcube è stato indirizzato verso lo sviluppo di un prodotto che avesse specifiche caratteristiche architetturali e funzionali

I principi seguiti nella progettazione sono quindi:

- Vantaggioso TCO (Total Cost of Ownership)
- Adesione agli standard di mercato (Protezione degli investimenti già effettuati)
- Integrabilità
- Disponibilità multiplatforma
- Possibilità di personalizzazione
- Semplicità di manutenzione
- Scalabilità



Principali ambiti applicativi

Le applicazioni per le quali la piattaforma RMcube è stata progettata sono principalmente:

- Sales Force Automation, ovvero sistemi per l’automazione della forza vendita
- Self Provisioning, ovvero sistemi di approvvigionamento in modalità “self service”
- CRM, ovvero Customer Relationship Management
- Ottimizzazione del processo di vendita, ovvero la gestione integrata delle trattative commerciali con stato, probabilità di chiusura, gestione documenti, valore, forecast e relativa reportistica. (To be developed)
- E-commerce
- Auctioning per la supply chain, ovvero la pubblicazione di aste verso i fornitori. (To be Developed)

Multicanalità

Da sempre le figure che contribuiscono alla catena di creazione del valore hanno operato in luoghi geografici diversi.

L’avvento delle tecnologie Wireless permette finalmente di poter portare l’informazione in tempo reale direttamente dove serve, cioè di fronte al cliente. Le tecnologie Wireless



necessitano di apparecchiature specificatamente progettate, quali i personal computer palmari. Piccoli, potenti, possiedono schermi di elevata qualità ma di piccola dimensione. E' fondamentale, per un impiego reale, che la suite applicativa possa impiegare uno strato di visualizzazione delle informazioni adattabile alle necessità e limitazioni che i dispositivi wireless impongono.

RMcube possiede la capacità innata di poter presentare i contenuti con insiemi differenti e contemporanei di templates xml ottimizzati in funzione del dispositivo di visualizzazione.

Internazionalizzazione

Comunicare significa anche farlo oltre i confini ormai ristretti delle nazioni. In una economia sempre di più globale anche adeguate capacità multilinguistiche nel software sono divenute imprescindibili.

RMcube fornisce supporto ad un numero virtualmente illimitato di lingue e garantisce una semplice manutenzione delle stesse anche in scenari nei quali si fa uso di un numero elevato di templates xml diversificati per gli utenti. Nella suite non solo infatti è fornito il supporto multilingue per gli utenti transattivi, ma anche per gli utenti amministrativi.

Scenari applicativi

I seguenti scenari possono essere soddisfatti tutti contemporaneamente e vengono descritti separatamente solo per comodità e semplicità di esposizione. Una unica applicazione RMcube può erogare contemporaneamente servizi differenziati ai vari utenti con estrema semplicità.

Sales Force Automation

In uno scenario di Sales Force Automation, gli Agenti possono interagire con RMcube in due modalità.

Dal proprio desktop, a casa o in ufficio, mediante una connessione Internet anche di tipo gratuito, potranno loggarsi e verificare la situazione di un cliente, inserire un nuovo ordine, visualizzare le caratteristiche dei nuovi prodotti.

Tramite un computer palmare dotato di accesso internet (wireless tramite GSM o GPRS) potranno impiegare RMcube per inputare direttamente l'ordine presso il cliente, verificarne il fido o controllare le spedizioni della merce acquistata.

Per questo sarà impiegato un apposito set di templates XML appositamente progettati ed ottimizzati che potrà essere scelto anche immediatamente dopo la procedura di login. L'Agente seleziona il cliente dal proprio portafoglio che viene immediatamente popolato dopo la procedura di autenticazione. Alla scelta del cliente, le condizioni commerciali dello stesso vengono utilizzate da RMcube per la gestione della distinta di ordine. Listini e sconti, condizioni di pagamento e dati anagrafici vengono così resi disponibili all'Agente per operare.

L'impiego del sistema di filtri di RMcube permette anche di popolare un catalogo dinamico dedicato al cliente specifico, con articoli o promozioni disponibili in esclusiva. Sono previste due applicazioni aggiuntive a supporto di questa operatività, la gestione delle RFQ, ovvero delle richieste di quotazione e la gestione delle vendite, ovvero la gestione delle informazioni sui prospects, sulle probabilità di vendita, con generazione del forecast.

Self Provisioning

La funzionalità di Self Provisioning offre al cliente la possibilità di inviare un ordine in modalità Self Service. In questo caso, dopo la procedura di login, il cliente viene accolto direttamente all'interno di un ambiente web che concentra tutte le informazioni che lo riguardano. Accesso allo storico ordini, visualizzazione dello stato di uno specifico ordine, disponibilità della merce.

Mediante la funzionalità di basket può essere costituito ed inoltrato un nuovo ordine che



incorpora automaticamente tutti i dati anagrafici.

La costruzione della distinta di ordine procede in modalità semplificata mediante ricerche e navigazione all'interno di un catalogo illustrato nel quale le schede prodotto possono contenere anche documenti digitali quali manualistica, istruzioni di montaggio, tabelle di compatibilità, documentazione sulle caratteristiche tecniche, istruzioni di uso audiovideo. Gli ordini effettuati vengono mantenuti all'interno di un apposito repository nel quale un amministratore può visionarli, spuntarli ed inoltrarli verso il sistema informativo. In alternativa l'inoltro verso il sistema informativo può avvenire in maniera automatica. Tramite il sistema dei filtri è possibile personalizzare l'aspetto grafico e le informazioni presentate in una modalità one2one per ciascun cliente o gruppo di clienti.

Funzionalità

Descrizione generale dell'applicazione

RMcube implementa un sistema e-business web based multiwebstore, multilingua, multivaluta, multiutente, multicanale, multipiattaforma.

L'applicazione è basata su:

- un sistema di catalogo parametrico.
- un sistema di gruppi di utenti con privilegi operativi correlati.
- un sistema di autenticazione degli utenti.
- un sistema di filtri di accesso che operano su specifiche griglie
- un sistema modulare di gestione delle transazioni.
- un "carrello" collegato a ciascun "cliente", quindi alla "sessione" che lo rappresenta.
- un processo di monitoraggio e gestione delle sessioni http.
- un sistema per la gestione ed il calcolo delle spese di spedizione.
- un sistema per la gestione completa delle immagini basato su applet.
- un sistema di gestione e riciclo delle connessioni Jdbc a sorgenti dati multiple.
- un motore di interpretazione di templates XML/XHTML.
- un avanzato sistema di marcatori per la visualizzazione dei dati e delle variabili in modalità multilingua.

L'architettura impiega un unico servlet di controllo che è in grado di reindirizzare le richieste a classi logiche Java distinte e modulari ciascuna delle quali implementa delle funzioni operative specifiche e può accedere ai dati mediante i servizi di uno strato di connettività gestita che si prende carico di riciclare in maniera ottimizzata le connessioni Jdbc ai databases (Connection pool manager) Questo strato rende indipendente l'applicazione dall'application server impiegato permettendo di far girare RMcube all'interno di tutti i principali application servers di mercato.

Le classi logiche possono creare oggetti che possono contenere valori dei campi dal record set o variabili. Ciascun oggetto può essere rappresentato sui templates XML e quindi "renderizzato" mediante il motore di interpretazione dei templates.

L'applicazione può essere personalizzata sostituendo certe "azioni", ovvero le classi che le implementano, oppure ampliata scrivendone di nuove.

Per questo specifico task vengono forniti degli "scheletri" di classi per guidare lo sviluppo.

RMcube può operare con tutti i principali database di mercato, sia commerciali che open source, dotati di un driver Jdbc (o mediante il bridge Jdbc/odbc, dotati anche solo di un driver Odbc).

All'interno del database RMcube crea un proprio sistema di tabelle necessarie a contenere le strutture dati impiegate. Le tabelle di RMcube hanno tutte un nome che inizia con RM in maniera da limitare il rischio di conflitti all'interno di database complessi che già



ospitano strutture dati.

La struttura dati, soprattutto per alcune componenti, per esempio quella che supporta la parte di catalogo, sono complesse a causa della flessibilità e parametricità richiesta dalle stesse.

Multiwebstore: E' definito webstore la partizione logica all'interno dell'applicazione che realizza i servizi per un unico centro di fatturazione. All'interno di un webstore tutti i prodotti che vengono scelti terminano all'interno della stessa distinta di ordine. Un webstore ha i propri "clienti" autorizzati, uno o più amministratori, una propria anagrafica prodotti, i propri filtri di accesso, i propri templates xml multipli, i propri sistemi di transazione, le proprie valute, le proprie immagini.

Multilingua: RMcube è in grado di gestire all'interno di ciascun Webstore, un numero virtualmente illimitato di lingue. I templates XML/XHTML contengono un sistema per l'implementazione delle traduzioni che semplifica la manutenzione di applicazioni multilingua. La capacità di gestione di alfabeti particolari è subordinata all'impiego di Databases e drivers Jdbc con capacità Unicode nonchè all'impiego di browser e sets di caratteri specifici.

Multivaluta: RMcube possiede la gestione di valute multiple illimitate e permette di specificare, per ciascuna di esse, cambio, arrotondamento simbolo. Sui Templates possono essere visualizzate due valute contemporaneamente, la valuta di default del webstore, ovvero quella impiegata per le eventuali transazioni, ed una tra le valute informative, che viene valorizzata dinamicamente applicando il cambio inserito rispetto alla valuta di default.

Multiutente: RMcube prevede sostanzialmente 2 classi di utenti, quelli amministrativi e quelli transattivi. Quelli amministrativi svolgono le funzioni di amministrazione, ovvero controllano gli ordini, li stampano, li esportano verso il sistema informativo, modificano i filtri, il catalogo, i listini, le promozioni etc. Viceversa gli utenti transattivi impiegano il sistema per inviare ordini, per controllare le disponibilità, per visualizzare il catalogo, per controllare lo stato di un ordine, etc. All'interno di ciascuna classe di utenti, mediante il sistema di privilegi di accesso, i gruppi ed i filtri collegati ai gruppi è possibile implementare politiche complesse di accesso alle informazioni ed alle funzioni.

Multicanale: RMcube possiede nativamente la capacità di presentare le informazioni secondo sets di templates XML/XHTML multipli contemporanei. Questo permette di fare accedere ad un sistema RMcube gli utenti amministrativi e transattivi indipendentemente, tramite templates adatti a device differenti, browser PC, web television, Chiosco multimediale, Computer Palmare wireless etc. E' possibile anche tecnicamente immaginare che i templates XML possano essere processati tramite Stylesheets XLS ed essere trasformati dinamicamente.

Multiplatforma: RMcube opera all'interno della piattaforma Java. Il requisito minimo per operare è un servlet container simile a Tomcat di Apache. Java è disponibile per tutti i principali ambienti di elaborazione e quindi RMcube può operare sostanzialmente su tutti i principali sistemi operativi compresi Linux, Microsoft Windows nelle varie versioni, AIX, Solaris, MacOS X, etc, etc.

Definizione dei ruoli operativi sulla piattaforma

Per maggiore chiarezza nominiamo queste figure illustrandone brevemente il ruolo:



Gestore del sistema: Figura amministrativa con privilegi amministrativi massimi. Può sostanzialmente eseguire qualsiasi azione all'interno del sistema ed ha la responsabilità della creazione ed amministrazione delle altre figure amministrative.

Gestore di Webstore: Figura amministrativa creata dal Gestore del sistema. Può amministrare uno o più webstores

Utente Transattivo: Figura utente che impiega il sistema. Può essere un semplice visitatore, un Cliente, un Agente. Gli utenti transattivi si dividono in due grandi famiglie: Utenti transattivi anonimi, Utenti transattivi autenticati. I primi possono interagire con il sistema senza passare attraverso una fase di Login, i secondi devono obbligatoriamente eseguire il login sul sistema e passare la procedura di autenticazione basata sulla fornitura di uno UserID ed una Password associata.

Funzionalità a disposizione del gestore di sistema

Il gestore del sistema è l'account con i privilegi più grandi all'interno di un sistema RMcube.

In un sistema RMcube esiste un solo account di "gestore di sistema".

Il gestore di sistema imposta i parametri funzionali di "ambiente", ovvero le impostazioni aggiuntive alla configurazione di base preinstallata da Antica Bottega Digitale.

Nel dettaglio opera su:

- Creazione dei Webstores , che si configurano automaticamente secondo un profilo di default generico fornito di serie da Antica Bottega Digitale. Tutte le impostazioni sono naturalmente successivamente modificabili.

- Creazione di utenti amministrativi per i webstores

- Configurazione delle temporizzazioni delle sessioni http

- Creazione degli attributi articoli di ambiente

- Creazione delle valute di ambiente

- Creazione di formati immagini di ambiente

- Creazione di templates xml aggiuntivi per il front end per gli utenti transattivi

- Creazione di templates xml aggiuntivi per il front end amministrativo

- Creazione delle lingue aggiuntive per il front end degli utenti transattivi

- Creazione delle lingue aggiuntive per il front end amministrativo

- Creazione delle destinazioni geografiche per le spedizioni

- Creazione delle modalità transattive

- Creazione dei vettori per il trasporto e dei metodi di pagamento eventualmente associati

- Popolazione degli stati parametrici per gli ordini

Il gestore di sistema non può operare all'interno del singolo webstore.

Funzionalità a disposizione dei gestori di webstore

Il Gestore di Webstores è una figura amministrativa che viene creata dal Gestore di Sistema.

In ogni Webstores esiste almeno un Gestore di Webstore.

Più Webstores possono avere lo stesso Gestore di Webstore, ovvero un medesimo Gestore di Webstore può amministrare più Webstores.

Il Gestore di sistema può creare più Gestori di Webstores per ogni Webstore.

Il Gestore di Webstore esegue:

- Creazione di utenti amministrativi per i webstores di propria competenza. Tali utenti amministrativi aggiuntivi non possono creare altri amministratori, ma a loro possono essere delegate tutte le altre operazioni interne al webstore

- Creazione delle valute locali (cioè valide solo in quel webstore)

- Creazione degli attributi articoli locali



- Creazione di formati immagini locali
- Creazione di templates xml aggiuntivi per il front end per gli utenti transattivi locali
- Creazione di templates xml aggiuntivi per il front end amministrativo locali
- Creazione delle lingue aggiuntive per il front end degli utenti transattivi locali
- Creazione delle lingue aggiuntive per il front end amministrativo locali
- Creazione delle destinazioni geografiche per le spedizioni locali
- Creazione delle modalità transattive locali
- Creazione dei vettori per il trasporto locali
- Popolazione degli stati parametrici per gli ordini locale
- Creazione e gestione del catalogo
- Inserimento dei files multimediali a supporto della presentazione degli articoli
- Creazione e gestione dei filtri di accesso per gli utenti transattivi e per quelli amministrativi
- Creazione e gestione degli utenti autorizzati
- Creazione e gestione delle viste per gli utenti transattivi e per quelli amministrativi
- Creazione e gestione dei listini
- Creazione e gestione delle promozioni
- Creazione e gestione dei portafogli agenti

Funzionalità a disposizione degli utenti

Gli utenti transattivi operano sul sistema mediante un front end costituito da un insieme di templates xml

Gli utenti transattivi appartengono a gruppi.

Un gruppo può essere formato anche da un solo utente.

Inizialmente nel sistema è definito un unico gruppo per gli utenti transattivi, il gruppo di Default, valido sia per gli utenti transattivi anonimi che per quelli autenticati.

Le funzionalità a disposizione degli utenti possono variare in dipendenza del gruppo al quale appartengono.

A ciascun gruppo è associato un filtro, che è costituito da 4 griglie che operano su:

- Insieme degli articoli presenti nel catalogo visibili tramite il filtro
- Listino impiegato per il gruppo di utenti transattivi associato al filtro
- Scontistica e promozioni previsti per il gruppo associato al filtro
- Set di templates xml dedicati al gruppo

salvo specifiche restrinzioni legate al meccanismo di filtri, agli utenti è generalmente possibile:

Visualizzare un catalogo illustrato dei prodotti con possibilità di eseguire ricerche, navigare in categorie e sottocategorie secondo un sistema di livelli virtualmente illimitato

Inserire prodotti all'interno di una distinta di ordine specificando le quantità e gli attributi degli articoli

Navigare all'interno delle promozioni

Confermare e registrare una distinta di ordine

Eseguire una transazione on-line tramite payment gateway esterno

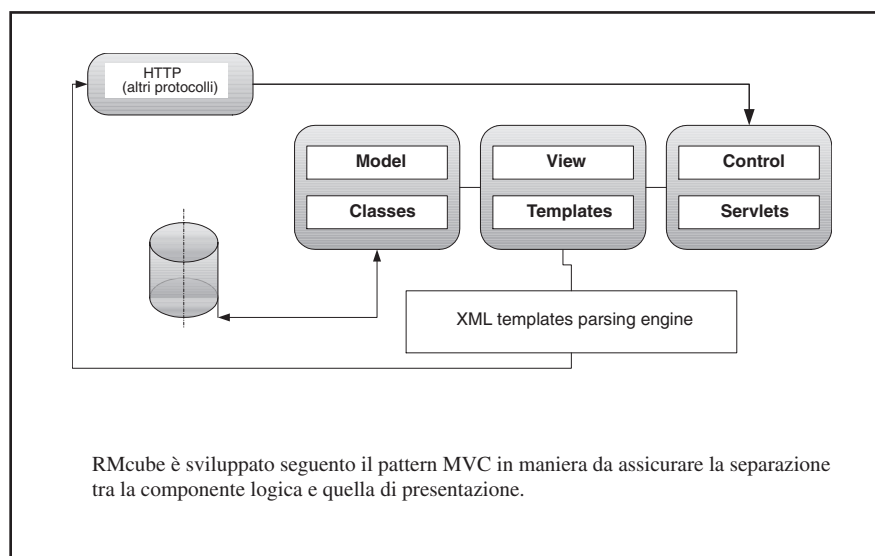
Controllare lo stato di avanzamento dei propri ordini effettuati.

Per gli agenti, è possibile selezionare dal portafoglio clienti il cliente specifico per il quale operare, eseguire ordini per lo stesso e visualizzarne la situazione ordini.



Cenni sulla architettura sistemistica

L'architettura della suite è organizzata secondo il modello MVC, Model View Control. Nello specifico, trattandosi di una applicazione Multitier che opera su web, il Model



è costituito da una serie di classi java che incapsulano la logica applicativa. Il control è rappresentato da una o più servlet mentre la parte di view viene svolta da appositi templates xml.

La possibilità di scalare l'applicazione è una diretta conseguenza della sua natura multitier. I vari "livelli" applicativi possono operare su cpu differenti per una efficace distribuzione del carico.

Web server, application Server, Database server possono essere supportati ciascuno da uno o più servers multiprocessore.

Specifiche configurazioni hardware e software ridondate possono essere impiegate per garantire prestazioni fault tolerant.

RMcube utilizza il database di backend per mantenere una immagine della sessione http di ciascun utente. Questo meccanismo permette all'applicazione, in caso di failover di un server applicativo, di recuperare la transazione con uno specifico utente impiegando alla successiva richiesta un altro server applicativo presente nella server farm.

RMcube è costituito da un insieme di servlets di controllo in grado di operare all'interno di un servlet container standard, una serie di classi logiche, un componente per la gestione e il riciclo delle connessioni alla base dati, un motore di parsing dei templates, una o più collezioni di templates xhtml/xml corredati da una collezione di marcatori per i dati.

Il flusso logico/operativo è il seguente:

Al momento in cui viene eseguita una richiesta, questa viene passata ad un servlet di controllo. In funzione delle "actions" contenute, vengono richiamate dal servlet delle classi logiche per le operazioni. Le classi logiche possono accedere alla base dati mediante i servizi di una componente specifica per la gestione delle connessioni, chiamata "Connection pool manager". Il compito di questo strato di software è quello di eseguire la gestione dell'accesso ai dati a partire da un certo pool di connessioni JDBC preallocate e finite in maniera da ottimizzare i tempi di risposta evitando continue connessioni e disconnessioni dalla base dati. Il "Connection pool manager" è in grado di gestire connessioni contemporanee a database servers multipli e di fornire i servizi a tutte le classi che ne facciano richiesta, anche quelle eventualmente sviluppate da terze parti mediante una specifica API.

La presenza di questa componente assicura la indipendenza dell'applicazione da specifici servizi di questo tipo forniti dagli application servers J2EE.



Quando le classi logiche hanno eseguito l'interazione con il database, esse popolano una collezione di variabili per la rappresentazione dei dati ed eseguono una chiamata al motore di rendering invocando un template xml per la rappresentazione.

Il motore di rendering dei templates esegue il parsing e sostituisce i marcatori riconosciuti all'interno dello stesso tramite i corrispondenti valori delle variabili.

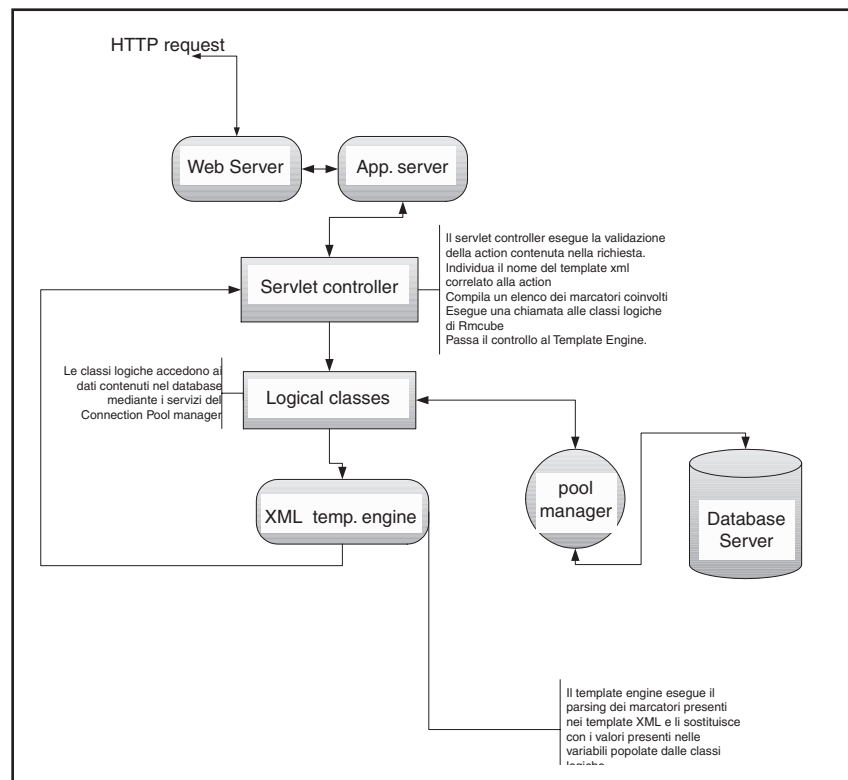
In ultimo il controllo dell'output è restituito al servlet per l'invio del tutto verso il browser che ha eseguito la richiesta.

Il sistema dei marcatori ed la struttura dei templates

RMcube supporta collezioni multiple di templates xml/xhtml secondo una logica che ne permette l'utilizzo contemporaneo.

Tale scelta è legata alla esigenza di poter sviluppare soluzioni multicanale o che necessitano di presentare informazioni diverse ad utenti differenti o con soluzioni grafiche differenti.

I templates sono progettati per essere facilmente editabili con editor visuali e per poter gestire virtualmente un numero illimitato di lingue in maniera razionale.



Essi mettono anche a disposizione un semplice linguaggio di scripting utile per particolari scopi e per modificare localmente i templates stessi in maniera dinamica testando certe condizioni. Per esempio tramite tale scripting è possibile eseguire un test sul fatto se l'utente è un utente loggato o meno e far comparire la funzione di login oppure il pulsante per eseguire il logout. I templates possono seguire una sintassi conforme allo standard xml. I comandi di scripting più comunemente usati sono `<if "condizione"> </if>`, `<list "oggettovettore" as element></list>` `<include "url del template da includere"> </include>` Che rispettivamente sono in grado di testare una condizione presentare una sezione html



o un'altra, costruire strutture tabellari mediante cicli e includere sezioni esterne di html in un template. Per informazioni in dettaglio si rimanda ad un approfondimento tecnico.

Estratto da Template: Visualizzazione del comando “refresh” all’interno di un form

//Esegue il codice se esiste la funzione viewShoppingCart e ne alloca i contenuti

`<if viewShoppingCart>`

//Imposta il ciclo da eseguire per tutti gli elementi che compongono l’oggetto

`<list cart.getObjectVector().iterator as element>`

//Visualizza il form che richiamerà il refresh

`<form action="{absolutePathServlet}/{controller}" method=POST>`

`<td>`

//Identificativo del template da caricare associato alla funzione chiamata

`<input type="HIDDEN" value =“viewShoppingCart” name = “function”>`

//Identificativo del comando da eseguire

`<input type="HIDDEN" value =“refresh” name = “command”>`

//Variabili passate alla funzione

`<input type="HIDDEN" value =“{element.getIdElement()}” name = “idElement”>`

//Variabili passate alla funzione

`<input type="HIDDEN" value =“codeShop” name = “{codeShop}”>`

`</td>`

`</form>`

`</list>`

`</if>`



Il sistema di gestione delle traduzioni

```
<function allocateLanguage()>

<switch lang>

<case "EN">
  <assign basketmessage ="This is your shopping basket">
  <assign itemcode ="Code">
  <assign variant = "VARIANT">
  <assign desc1 = "From Product" >
  <assign desc2 = "Product Description : " >
  <assign desc3 = "Variant Description : " >
  <assign desc4 = "Stock Information : " >
  <assign desc5 = "Stock : " >
  <assign desc6 = "Price : " >
  <assign desc7 = "Price : " >
  <assign desc8 = "No sellable countries : " >
  <assign desc9 = "Quantity" >
  <assign desc10 = "Erase" >
  <assign desc11 = "Update" >
  <assign desc12 = "Cost" >
<break>

<case "IT">
  <assign basketmessage ="Nel vostro carrello sono presenti i seguenti prodotti">
  <assign variant = "VARIANTE">
  <assign itemcode ="Codice">
  <assign desc1 = "Appartenente al prodotto" >
  <assign desc2 = "Descrizione Prodotto : " >
  <assign desc3 = "Descrizione Variante : " >
  <assign desc4 = "Informazioni sulla Giacenza : " >
  <assign desc5 = "Giacenza : " >
  <assign desc6 = "Prezzo : " >
  <assign desc7 = "Prezzo : " >
  <assign desc8 = "Non è vendibile nei seguenti paesi : " >
  <assign desc9 = "Quantità" >
  <assign desc10 = "Cancella" >
  <assign desc11 = "Aggiorna" >
  <assign desc12 = "Totale" >
  <break>
<default>
<html>
<title>Shopping Cart</title>
<base href="{absolutePathTemplate}/{relativePath}">
<link rel="stylesheet" href="css/robov3.css" type="text/css">
<body bgcolor="#FFFFFF" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0">
<include includePath+"/included_testata_template.html"></include>
<!-- qui c'è un include -->
<table width="100%">
<tr>
<td width="10%" valign="top"><include includePath+"/included_login_template.html"></include><!-- altro
include per il login --></td>
<td valign="top"><if emptyCart>
<div align="center">
<p><b><font size="+1" face="Verdana, Arial, Helvetica, sans-serif">${desc24}</font></b>
</p>
</div>
</if>
<div align="center"><font face="Verdana, Arial, Helvetica, sans-serif" size="+1"></font><font
size="+1"></font><if shoppingCart>
</if></div>
<if shoppingCart>
<p align="center"><font face="Verdana, Arial, Helvetica, sans-serif"
size="+1"><b>${basketmessage}</b></font></p>
<div align="center">
<table border=0 bordercolor="#B5BCD6" cellpadding=4 cellspacing=1 class="Rmv3testobase" align="center"
bgcolor="#FFFFFF">
<tr valign=TOP>
<th bgcolor="B5BCD6"><b>${itemcode}</b></th>
<th bgcolor="B5BCD6">
<p><b>${desc22}</b></p>
</th>
</tr>
</table>
</div>
</if>
</table>
</body>
</html>
```

Sezione di definizione della lingua Italiano.

La funzione `allocateLanguage()` esegue uno switch impiegando una sezione o un'altra in funzione della lingua selezionata dall'utente sul frontend.

E' possibile aggiungere liberamente altre sezioni lingua, variabili e le rispettive traduzioni.

Con questo meccanismo è possibile gestire la traduzione multilingua con un solo set di templates semplificando molto la manutenzione e le successive modifiche grafiche.

Il motore di rendering dei templates sostituisce le stringhe di traduzione ai marcatori impiegati all'interno dei templates nelle loro rispettive posizioni, come mostra la linea tratteggiata per il marcatore `${itemcode}` che viene sostituito con la stringa `Code` se l'utente ha impostato la lingua inglese per la navigazione sul front end



Conclusioni

Rmcube è una suite applicativa sviluppata con criteri di eccellenza tecnica, adatta ad implementazioni che utilizzino anche componenti open source ma in grado di operare con tutti i principali standard di mercato..

La architettura tecnica basata su un approccio modulare permette alla clientela di aggiungere con semplicità le proprie funzioni utilizzando le API disponibili ed integrare così completamente la suite all'interno dei sistemi informativi.

Scalabilità e robustezza sono garantite dalla adesione alla piattaforma J2EE.

Queste caratteristiche candidano RMcube non solo al ruolo di prodotto applicativo per progetti di Sales Force Automation, E-commerce o Self Provisioning, ma anche ad essere impiegato come ambiente di sviluppo per ampi progetti di CRM che utilizzino elettivamente il web quale infrastruttura di comunicazione con i propri partners e con la propria clientela e che necessitano di una rapida integrazione del front end con i sistemi informativi esistenti.





ANTICA BOTTEGA DIGITALE
Person e & Tecnologie

Antica Bottega Digitale srl
Via Bologna 14 - 52100 Arezzo
Tel. 0575 32671
Fax 0575 3267999
<http://www.abd.it>