

Autore: Nicola CECCON

Classe: QUINTA INFORMATICA (5IA)

Anno scolastico: 2004/2005

Scuola: Itis Euganeo

APPLICAZIONE WEB

Sintesi sulla tematica della progettazione e realizzazione di applicazioni web, cioè di quelle applicazioni che permettono la creazione di siti dinamici, mediante l'utilizzo delle tecnologie internet, per la realizzazione di servizi interattivi dipendenti dalla consultazione di database remoti.

Come applicazione web di tutorial si presenta un tipico sito per il commercio elettronico.

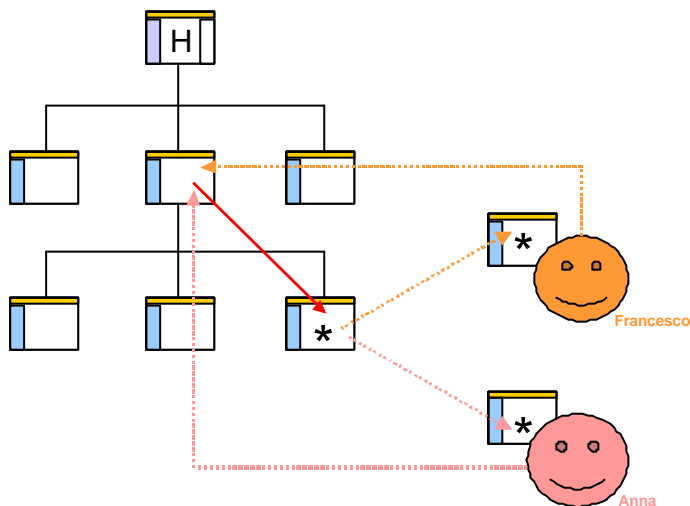


Basic concepts

E' fondamentale capire che nella interazione con un sito non necessariamente quello che è memorizzato sul server web è quello che l'utente visualizza sul proprio browser, in altri termini il file sul server può essere diverso dal contenuto trasmesso. Questa differenza è legata al fatto che l'attuale tecnologia permette di avere pagine web statiche o dinamiche.

Pagina statica

In una pagina web statica, il contenuto della pagina viene stabilito nel momento in cui si crea e si memorizza la pagina sul server web e quindi ogni volta che un utente accede a una pagina statica, questa gli presenta le stesse informazioni, a meno che non venga aggiornata, cioè riscritta.

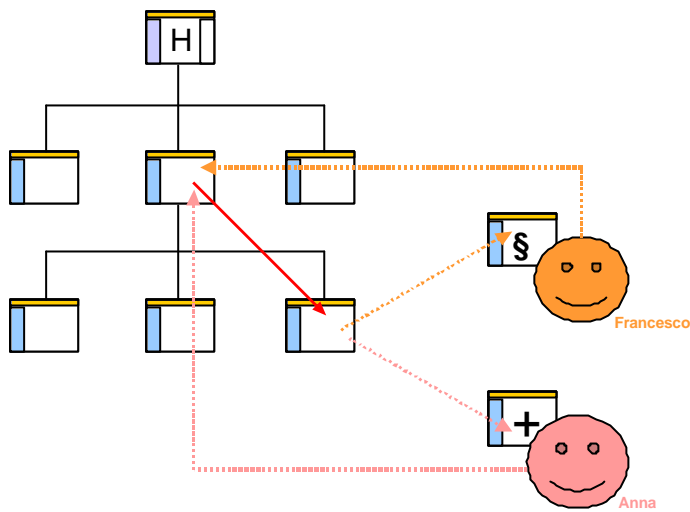


Per esempio, gli utenti Francesco e Anna accedono alla pagina con il contenuto "" da un'altra pagina del sito e ricevono entrambi lo stessa pagina con il contenuto "*".*

Le pagine statiche sono pertanto indicate per visualizzare informazioni che non cambiano spesso e sono file HTML.

Pagina dinamica

In una pagina web dinamica, detta anche pagina web interattiva, i contenuti della pagina variano in funzione degli input dell'utente (per esempio l'autenticazione ad un servizio di mailing list) o del verificarsi di un evento sul server (per esempio il variare di una immagine nella Home page al variare dell'accesso).



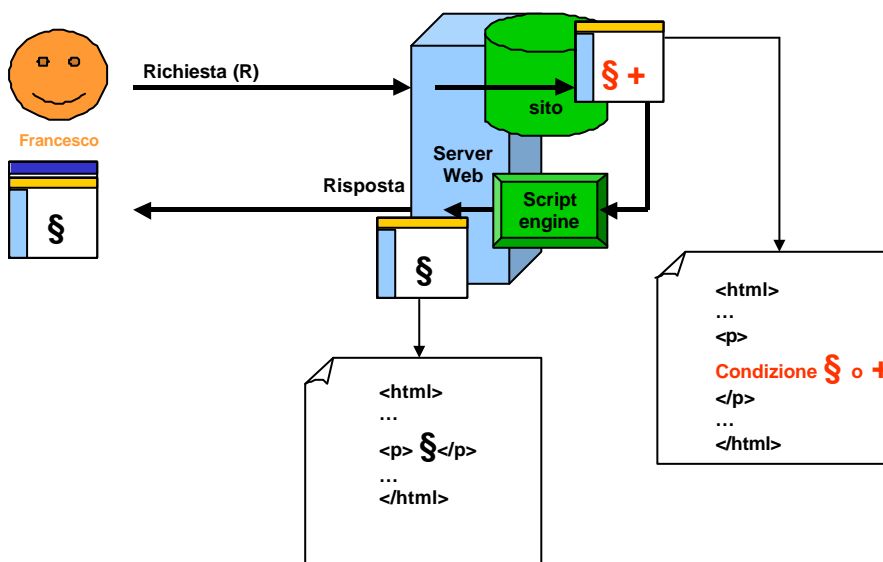
Per esempio, gli utenti Francesco e Anna accedono ad una pagina del sito provenienti da un'altra e ricevono due contenuti (pagine) diverse: Francesco riceve “\$” mentre Anna riceve “+”.

E' importante sottolineare che il codice della pagina dinamica è nascosto all'utente che invece riceve sempre una pagina HTML, cioè il **server web prima di inviare la pagina all'utente trasforma la pagina dinamica in una pagina statica** generando, come si dice al volo, il codice HTML come risultato di una elaborazione del codice dinamico presente nella pagina sul server.

Codice visto dall'utente

Questa elaborazione del codice dinamico è effettuata da un apposito modulo o componente software del server web o in stretta relazione con esso, definito **scripting engine**, che assolve alla funzione di **interprete del linguaggio dinamico**.

Script engine



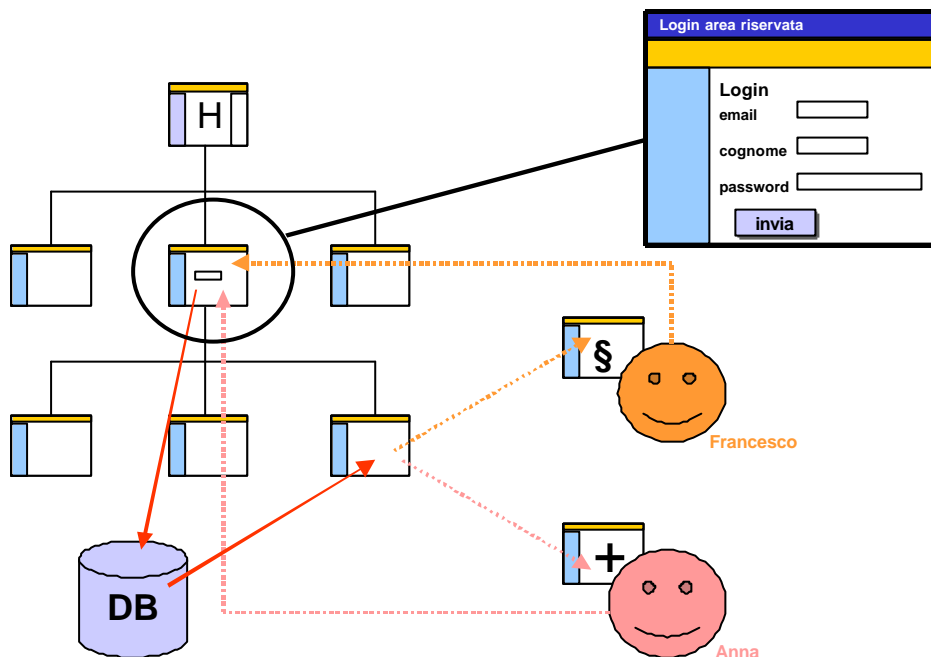
Per esempio Francesco richiede una pagina dinamica e il server web restituisce una pagina con solo codice HTML che visualizza il contenuto “\$” invece di “+” poiché è stato riconosciuto (condizione) dall'applicazione.

Database

I database e il server DBMS che li gestiscono giocano un ruolo fondamentale nella progettazione e costruzione di una applicazione web, infatti una tipica pagina web dinamica interroga un database (remoto) e visualizza i dati: queste richieste dell'utente utilizzano un sito, pilotato da database, con pagine web dinamiche.

Ti pi ca pagi na di nami ca

E' importante sottolineare che il server web e il server di database possono essere processi in esecuzione su una unica macchina come su macchine separate¹, collegate tipicamente, ma non necessariamente², in rete locale LAN.



Per esempio, gli utenti Francesco e Anna utilizzano un form per autenticarsi ed accedere ad un'area riservata del sito, i dati per l'autenticazione sono registrati in database. A seguito dell'autenticazione Francesco riceve come contenuto "\$" mentre Anna riceve "+".

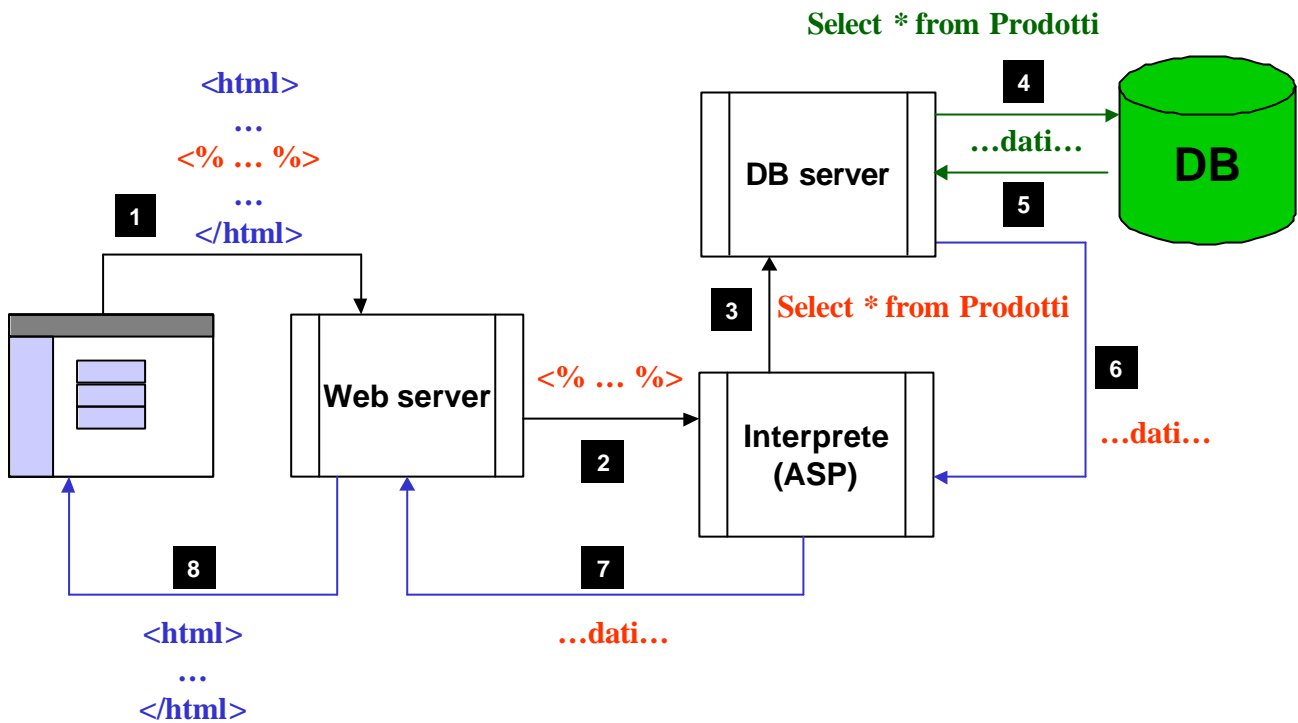
Di solito l'accesso ad una pagina dinamica mediante un form comporta, nell'ordine, i seguenti passi:

- Richiesta della pagina contenente il form per l'invio dei dati al server web.
- Compilazione del form, invio dei dati e quindi richiesta implicita di una pagina dinamica.
- Elaborazione della pagina dinamica da parte dell'interprete.
- Richiesta di dati al server di database.

¹ Ovviamente la scelta della separazione su due macchine server diverse è una soluzione più performante.

² Potrebbero essere collegate in modalità sicura attraverso internet.

- Invio dei dati all'interprete.
- Invio dei dati al server web.
- Creazione della pagina HTML alla fine della elaborazione.
- Invio al browser della pagina HTML.
- Visualizzazione della pagina HTML.



Quindi, l'utente, per prima cosa, chiede una pagina web dinamica e il browser dell'utente inoltra la richiesta al server web. Quindi, il server web avvia un programma per elaborare la richiesta.

Il programma compone una interrogazione (query) e la invia al database.

Il database restituisce al programma i dati richiesti. Qui i dati vengono formattati in una pagina web HTML dal server web. Poi il server rimanda la pagina web dinamica al browser dell'utente.

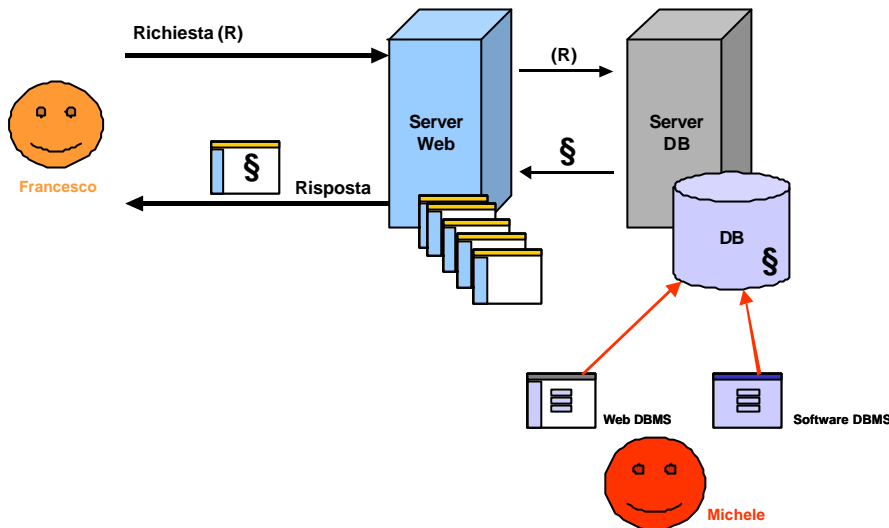
Tipologie di utenti e database

E' evidente che la presenza di un database comporta la definizione di tre tipologie di utenti:

- **Amministratore** che ha pieno accesso al database. Il database può essere modificato dall'amministratore tramite una o più pagine dinamiche, che costituiscono la parte della applicazione web riservata all'amministratore³, mediante il DBMS associato oppure un software appositamente scritto, per esempio in Visual Basic.

³ Questa parte prende il nome di Pannello di controllo.

- **Utente che ha un accesso parziale** spesso in solo lettura, tipicamente per visualizzare dati a seguito di query di ricerca.
- **Utente che ha un accesso anche in scrittura**, tipicamente per l'inserimento di dati che lo riguardano o l'utilizzo di alcuni servizi come l'inserimento di un prodotto nel carrello della spesa.



Per esempio Francesco (utente con accesso parziale) interagisce con il sito dinamico ricevendo “\$” mentre Michele (amministratore) gestisce il database utilizzando un DBMS e/o un pannello di controllo di gestione del database (web DBMS).

Quindi, l'interazione dipende dalla tipologia di utente e l'interazione non consiste solo in visualizzazione di dati, ma anche nella modifica o inserimento: si possono usare pagine web dinamiche anche per modificare il contenuto del database in funzione degli input dell'utente.

Interazione dipende dal tipo di utente

All'utente viene presentato un modulo o form, per consentirgli di immettere i dati che verranno impiegati per inserire nuovi record, oppure per modificare o per eliminare record esistenti.

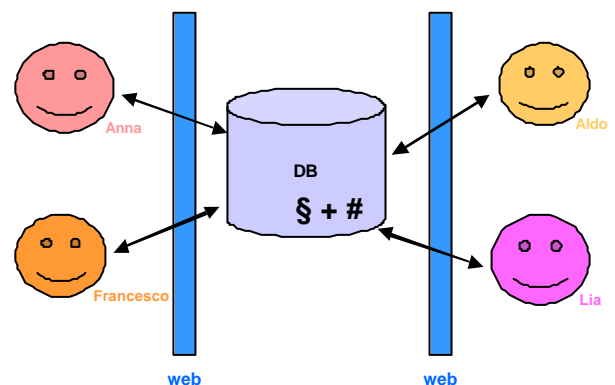
Definizione di applicazione web

Una applicazione web è costituita da pagine dinamiche.

Ambiente di una applicazione web

Iniziamo con la considerazione che si tratta di realizzare programmi per ambienti multiutente, cioè programmi che devono interagire contemporaneamente con più utenti, quindi programmare pagine dinamiche significa **anche** programmare per un ambiente multiutente.

Questo significa che indipendentemente dalla piattaforma (software per una rete o sito web) le



problematiche sono le stesse: il problema della integrità e della consistenza dei dati.

Ma la cosa più importante è capire la differenza tra la programmazione di una applicazione multiutente e la programmazione di una applicazione web.

La differenza non è data tanto dal fatto che il programma è distribuito tra pagine dinamiche, perché un software si realizza in moduli che interagiscono, ma dal fatto che il protocollo HTTP comporta che le pagine dinamiche **siano senza memoria**⁴.

Senza memoria significa che una volta che la pagina dinamica è stata processata dal server web, trasformata in HTML e inviata, tutti i dati della pagina vengono persi, cioè il server non mantiene le informazioni sulle precedenti richieste dell'utente o in altri termini il server web si stacca completamente dal client e si pone semplicemente in attesa di un'altra richiesta chiudendo la connessione con il client. Non vengono mantenute quindi le informazioni sullo stato della connessione del client e per questo si dice che il protocollo HTTP è stateless, cioè senza stato.

Senza memoria a causa del protocollo http che è stateless

Per esempio, se invio il codice personale alla pagina di autenticazione di un sito di ecommerce questo viene perso una volta che la pagina di autenticazione viene processata con l'invio della risposta dell'autenticazione: il codice non è più disponibile per le altre pagine, come la pagina per il pagamento con la carta di credito che necessita degli stessi dati.

Un'altra differenza rispetto ad altre applicazioni è data dalla larghezza di banda della trasmissione disponibile e dal carico di lavoro sul server.

Larghezza di banda

Questi due fattori incidono sulla quantità di informazioni trasferibili tra server e client ed anche sull'interfaccia utente che deve necessariamente essere alleggerita dai componenti grafici. E' quindi necessario, quando possibile, ottimizzare il carico di lavoro tra client e server facendo per esempio svolgere al client la validazione dei dati in modo da diminuire il carico sul server e la trasmissione di informazioni di controllo ed errore.

Definizione accurata

Una applicazione web è una applicazione client/server per un ambiente stateless, cioè senza memoria, che utilizza le tecnologie internet.

Pertanto, saper programmare per il web significa essere in grado di programmare in un ambiente "senza memoria", cioè conoscere i diversi meccanismi e strumenti per conservare o passare i dati, detti parametri, tra le diverse pagine dell'applicazione web.

⁴ Il modello di interazione del web è privo di del concetto di connessione, per cui l'esecuzione di una applicazione web consiste in una serie di interazioni disconnesse e compito del programmatore è garantire la continuità dell'applicazione durante la sequenza di interazioni dell'utente.

Per esempio, come fare in modo che il codice personale inviato alla pagina di autenticazione di un sito di ecommerce sia anche disponibile alla pagina per il pagamento con la carta di credito, senza dover reintrodurlo.

L'applicazione web si sostanzia in un sito realizzato con una o più tecnologie che permettono la creazione di un sito dinamico, cioè di un sito nel quale il contenuto delle pagine varia durante l'interazione.

Applicazione web
uguale a sito
dinamico

La variabilità dei contenuti di una applicazione web comporta una attenta progettazione di tutti gli elementi di un sito, anche se una particolare attenzione deve essere rivolta alla parte "software", cioè al codice che permette la dinamicità.

Approcci per creare pagine web dinamiche

Le pagine web dinamiche si possono creare mediante un'elaborazione dal lato server, oppure tramite una combinazione di elaborazioni dal lato server e dal lato client.

Nell'elaborazione dal lato server, il server web riceve la richiesta della pagina web dinamica, esegue tutte le operazioni necessarie per crearla e invia la pagina completata al client, affinché venga visualizzata nel suo browser.

Lato server

Nell'elaborazione dal lato client, alcune elaborazioni sono eseguite nella stazione di lavoro client, sia per comporre la richiesta della pagina web dinamica, sia per creare o per visualizzarla.

Lato client

Tecnologie di elaborazione

Le tecnologie utilizzate nelle elaborazioni dal lato server e dal lato client, sono le seguenti:

- Elaborazioni dal lato client
 - Scaricamento ed esecuzione nel client di programmi compilati
 - Script dal lato client
- Elaborazioni dal lato server
 - Programmi compilati
 - Programmi per servizi ISAPI/NSAPI
 - Script dal lato server

Elaborazioni dal lato client

Un approccio all'elaborazione dal lato client prevede lo scaricamento di programmi eseguibili compilati, conservati sul server web, nel browser dell'utente, e quindi la loro esecuzione nella stazione di lavoro dell'utente.

Programmi
eseguibili

Il browser e il sistema operativo dell'utente devono essere in grado di elaborare il file del programma eseguibile, devono cioè essere dotati di uno script engine. Questo programma interagisce con l'utente e, quando necessario, invia e reperisce dati da un server di database.

Esempi di tecnologie che usano questo approccio sono:

- **Applet Java.** Java è un linguaggio di programmazione che costituisce un sottoinsieme semplificato del C++. Si usa comunemente per creare applicazioni web, chiamate applet Java, che si possono scaricare nel browser di un utente da un server web, e quindi eseguire direttamente all'interno del browser. Dal momento che l'applet Java gira in un ambiente di esecuzione Java generico, fornito dal browser⁵, l'esecuzione dell'applet è identica in qualsiasi sistema operativo e con qualsiasi browser web. Per ragioni di sicurezza, le applet Java possono ricevere dati da un server web e inviarglieli, ma non possono leggere o scrivere dati su qualsiasi file nella stazione di lavoro dell'utente.
- **ActiveX** di Microsoft. I programmi ActiveX si possono creare mediante molti linguaggi di programmazione e in ambienti diversi, purché questi linguaggi supportino i protocolli ActiveX della Microsoft.

La differenza principale fra un programma web ActiveX e un'applet Java risiede nel modello della sicurezza, infatti le applet Java hanno un rigido modello della sicurezza, che impedisce loro di danneggiare il computer client sul quale operano mentre le applicazioni ActiveX hanno accesso completo⁶ al sistema operativo e ai file del computer client.

Un altro approccio all'elaborazione dal lato client si basa sugli script⁷.

Client script

Questa tecnica permette di digitare codice non compilato, in linguaggi quali JavaScript⁸ o VBScript, nel documento HTML insieme con il testo statico HTML: Tag (marcatori) particolari indicano al browser dell'utente che il testo è un codice di programma. Se il browser è in grado di riconoscere e di interpretare il codice, lo esegue lo visualizza come testo.

Grazie a questo approccio, è possibile realizzare *Tag* interfacce utente più complesse di quelle che si possono ottenere tramite il normale HTML⁹.

Inoltre, questa tecnica consente la validazione¹⁰ dell'input, cioè di controllare la correttezza degli input dell'utente presso la sua stazione di lavoro, anziché nel server web, riducendo quindi il traffico di rete e

```
<html>
<script language="JavaScript">
function Stampa()
{
msg=window.print();
}
//-->
</script>
<body>
...
```

Funzione Javascript che implementa la stampa della pagina

⁵ Java Virtual Machine.

⁶ Per questo esiste una certificazione di riconoscimento e affidabilità della fonte del programma.

⁷ Uno script è un programma e la differenza fra i due termini è che per tradizione, i documenti eseguibili compilati sono chiamati programmi, e i programmi interpretati sono chiamati script.

⁸ Da non confondere con Java.

⁹ E' importante sottolineare la non stabilità degli script il cui funzionamento varia al variare del browser e delle versioni dello stesso browser. Inoltre, questo codice comporta dei problemi di accessibilità.

¹⁰ Il caso tipico è il controllo sulla presenza e sulla forma dell'indirizzo di posta elettronica in una casella di testa in un form.

migliorando le prestazioni dell'applicazione.

Elaborazioni dal lato server

Le elaborazioni lato server possono essere ottenute con programmi compilati eseguibili oppure programmi non compilati detti script che vengono interpretati ed elaborati durante l'esecuzione sul server.

I programmi compilati possono essere realizzati con diversi linguaggi¹¹ e indipendentemente dal linguaggio utilizzato il protocollo Common Gateway Interface **CGI** è il metodo per comunicare¹² tra la pagina HTML e il programma: è in pratica un protocollo per eseguire programmi su un server HTTP.

Uno svantaggio, nell'usare programmi basati su CGI, è che un server web avvia una copia del programma ogni volta che c'è la richiesta, quindi per esempio è possibile che un server web molto attivo esaurisca la sua capacità di memoria, se serve molti form contemporaneamente. Per questo motivo in seguito della diffusione di siti web interattivi, i fornitori di server hanno sviluppato tecnologie proprietarie. Fra gli esempi di tecnologie che risolvono questo problema, vi sono:

- Netscape Service Application Programming Interface (NSAPI, interfaccia di programmazione per applicazioni di servizio della Netscape).
- Internet Server Application Programming Interface (ISAPI, interfaccia di programmazione per applicazioni di server Internet), della Microsoft.

Script lato server

Un altro approccio per creare pagine web dinamiche che usano l'elaborazione dal lato server utilizza script dal lato server.

Uno script dal lato server è costituito da **codice non compilato** che viene inserito all'interno del file HTML di una pagina web, per estenderne le capacità, come per esempio, per abilitare il file HTML della pagina web a eseguire query sul database. Fra gli esempi¹³ della tecnologia che usa questo approccio, vi sono:

- Server-side includes SSI.

¹¹ Tipici linguaggi sono Java, Visual Basic ,C++ e soprattutto il Perl. Il Perl (Practical Extraction and Report Language) è un linguaggio finalizzato principalmente alla trattazione di stringhe e file di testo particolarmente per le ricerche di sequenze di caratteri all'interno di stringhe (pattern matching), sostituzioni di sottostringhe (pattern substitution), operazioni su file di testo strutturati in campi e record oppure non strutturati. Per questi motivi il Perl è abbastanza, anche se meno di un tempo, utilizzato nella scrittura di procedure CGI installate su un server web, o per lo sviluppo di procedure di manutenzione delle attività di un server.

¹² La tecnologia più diffusa per le pagine web dinamiche dal lato server utilizza form HTML, ovvero documenti HTML potenziati, progettati sullo stesso schema dei moduli cartacei, per raccogliere input dell'utente e quindi inviarli al server web.

¹³ Elenco non esaustivo.

- Active Server Page ASP¹⁴ di Microsoft.
- PHP.
- Java Server Pages JSP.
- Python¹⁵

Quando il browser di un utente richiede una pagina web che contiene uno script viene interpretato ed elaborato nel server web, e i dati risultanti sono memorizzati sulla pagina web che viene restituita al browser dell'utente.

Tag

```
<html>
<body>
<%
strQ="SELECT count (*) As Num FROM Prodotti"
connDB.Execute(strQ)
strMes = "Prodotti presenti" & Num
Respose.Write (strMes)
%>
...
</body>
</html>
```

Codice ASP che conta
il numero di record nella
tabella Prodotti

Architettura di una applicazione web

L'architettura di una applicazione web si sviluppa su tre livelli logico-funzionali, detta Three-Tier:

1. **Livello di presentazione.** È il livello dell'interfaccia utente dell'applicazione web corrispondente al client ed è costituito da diversi componenti combinati tra loro come il browser, documenti HTML, applet Java, controlli Activex. Questo livello si occupa di acquisire dati e visualizzare i risultati.
2. **Livello intermedio,** detto Business Logic. È il livello che contiene la logica elaborativa dell'applicazione che permette di soddisfare le richieste di dati e di elaborazione del client. Il livello intermedio dipende dal server web utilizzato, e dai suo componenti, ed è costituito da script, cioè da codice dinamico, in base alla tecnologia utilizzata (CGI, ASP, ISAPI...).
3. **Livello dati** E' il livello che non fornisce servizi direttamente tramite il server web, ma che sono forniti da applicazioni indipendenti

¹⁴ Microsoft © Active Server Pages (ASP) is a server-side scripting environment that you can use to create and run dynamic, interactive Web server applications.

Dalla definizione di Microsoft si capisce che ASP non è un linguaggio di scripting, ma un ambiente per la creazione dinamica di pagine, cioè per la creazione di siti dinamici progettato e implementato da Microsoft per creare pagine web dinamiche che agisce principalmente sul lato server.

La generazione avviene con l'ausilio di un linguaggio di scripting ASP-compatibile, cioè supportato da ASP o meglio con più linguaggi dato che una pagina ASP può ospitare contemporaneamente diversi linguaggi di scripting. La pagina ASP contiene normalmente scripting ASP ed eventualmente codice nei linguaggi di scripting lato client (tipicamente JavaScript per la portabilità del codice su più piattaforme).

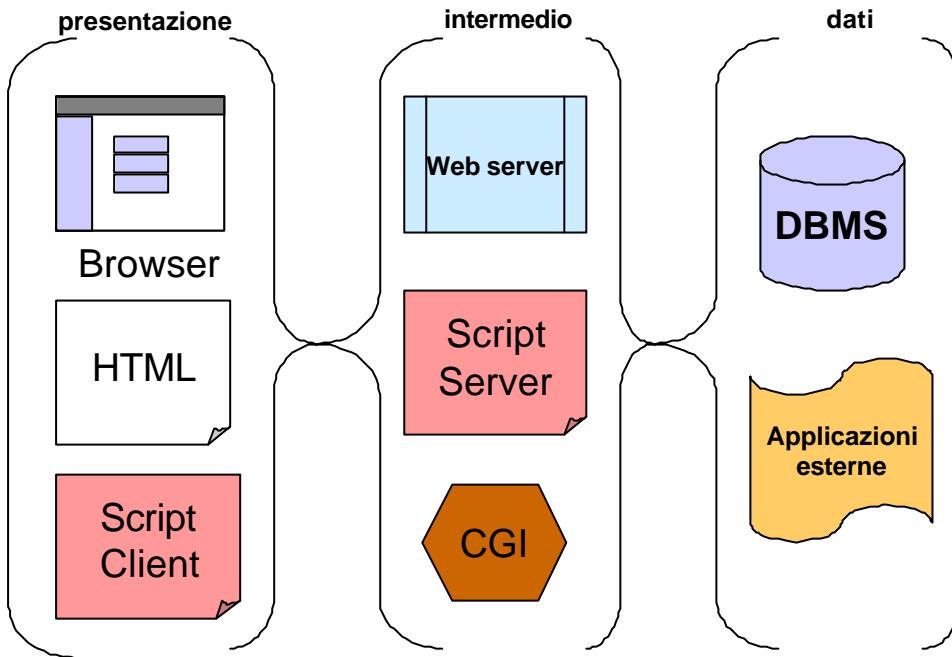
L'utilizzo di ASP implica la definizione di un file ".asp" che contiene:

- Testo
- Comandi ASP
- Linguaggi di scripting ASP-compatibili lato server
- Linguaggi di scripting lato client
- Tag HTML

In termini insiemistici si potrebbe definire l'insieme ASP in questo modo: ASP = (testo, tag HTML, comandi ASP, linguaggi di scripting).

¹⁵ In particolare viene utilizzato come linguaggio di attraverso la metodologia CGI.

dall'ambiente web. Tipici esempi sono i server di dati (DBMS) e i server di email. In particolare per i database è opportuno utilizzare delle tecnologie di connessione¹⁶ tra il livello intermedio e il livello dei dati in modo da rendere indipendente l'applicazione dalla tipologia di dati utilizzati per future modifiche.



Infrastruttura web

L'infrastruttura web è costituita dal sistema operativo e dal server web utilizzato.

Attualmente dominano due infrastrutture competitive:

1. Sistema operativo Linux¹⁷ con server web Apache.
2. Sistema operativo Microsoft Server con server web Microsoft IIS.

E' importante sottolineare che la scelta dell'infrastruttura web incide anche sulla scelta del linguaggio di scripting e sul database¹⁸.

¹⁶ Come ODBC, ADO...

¹⁷ O in termini più generali sistemi Unix come Mac OS Server X della Apple.

¹⁸ Per finalità didattiche si citano i database MySQL, Access e Microsoft SQL che non esauriscono l'insieme delle soluzioni affidabili presenti sul mercato come Oracle, Interbase, DB2... Non è comunque raro trovare come database Microsoft Access nonostante le limitazioni che comporta. In linea generale, nel primo caso si opta per il linguaggio PHP e il database MySQL mentre nel secondo per ASP e il database Microsoft SQL oppure MySQL.

Progettazione

In un'applicazione web si devono progettare:

1. La **Business Logic**, cioè le operazioni consentite che determinano l'interazione tra utente e sito.
2. Il **database**, in pratica quali informazioni si conservano e come.
3. Il **sito**, vale a dire l'organizzazione e interfaccia delle pagine.
4. Gli **script** per ogni pagina, utilizzando un linguaggio di pseudoscripting.

Il risultato della progettazione è formalizzato con degli schemi che si utilizzano nell'implementazione dell'applicazione.

Progettazione della Business Logic

Nella progettazione del funzionamento dell'applicazione è consigliabile procedere con una metodologia che si basa sulla definizione di **scenari d'utilizzo relativi a diversi tipi d'utente**, allo scopo di individuare le possibili interazioni e, quindi, le risposte dell'applicazione.

Scenari d'utilizzo

L'obiettivo è individuare le possibili interazioni degli utenti, in modo da saturare lo spazio d'utilizzo introducendo funzionalità del sistema o vincoli d'uso: per esempio l'individuazione di un utente registrato ad un servizio rispetto ad uno non registrato.

Si procede definendo:

1. **Gli utenti** con le informazioni personali e gli obiettivi dell'utente (rispetto al sito). Per esempio un utente che può inserire dati rispetto ad uno che può solo visualizzare.

Utente	
Alice	<p>Informazioni personali</p> <p>Nome: Alice Età: 32 Stato civile: single Occupazione: Impiegata</p>
	<p>Obiettivi</p> <p>Appassionata di cinema da molti anni, ha iniziato collezionare film in formato VHS sia attraverso la registrazione dalla Tv sia attraverso l'acquisto di film originali.</p> <p>Partecipa al Forum in modo attivo ed è registrata da tempo.</p>

2. **Gli scenari generali d'interazione.** Con gli scenari si riescono a stabilire i diversi moduli software che l'applicazione deve presentare, per esempio in un sito d'ecommerce uno scenario per un utente registrato e un altro per uno non registrato.

In particolare una tecnica di progettazione prevede l'utilizzo del linguaggio UML¹⁹ dei casi d'uso, detti "Use Case", che prevede l'individuazione di attori e situazioni di un particolare scenario, mentre in questa dispensa si utilizzano degli appositi diagrammi, appositamente creati, più semplici.

3. **Task utente.** Definiti gli utenti si procede alla definizione dei task dell'utente, cioè le possibili azioni che l'utente può compiere mediante l'utilizzo di un diagramma²⁰ in cui si evidenziano oltre ai task anche la destinazione di un task stabilita dall'applicazione.

Consideriamo, com'esempio, l'applicazione per un Forum.

Un Forum di discussione permette agli utenti di un sito di discutere tra loro, scambiando informazioni, consigli, pareri...

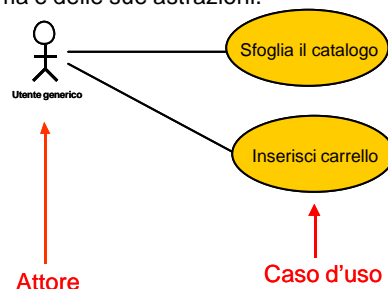
Forum

In un Forum i messaggi sono organizzati in maniera gerarchica, con in radice le discussioni e per ogni argomento i messaggi (da notare che sono possibili più livelli).

Un qualsiasi utente può leggere i messaggi (con diverse tecniche di selezione), ma solo gli utenti registrati al servizio possono inserire o rispondere (funzione Quote o Cita) ad un messaggio (mediante richiesta di dati personali e della coppia di informazioni per l'autenticazione: Nickname-Password).

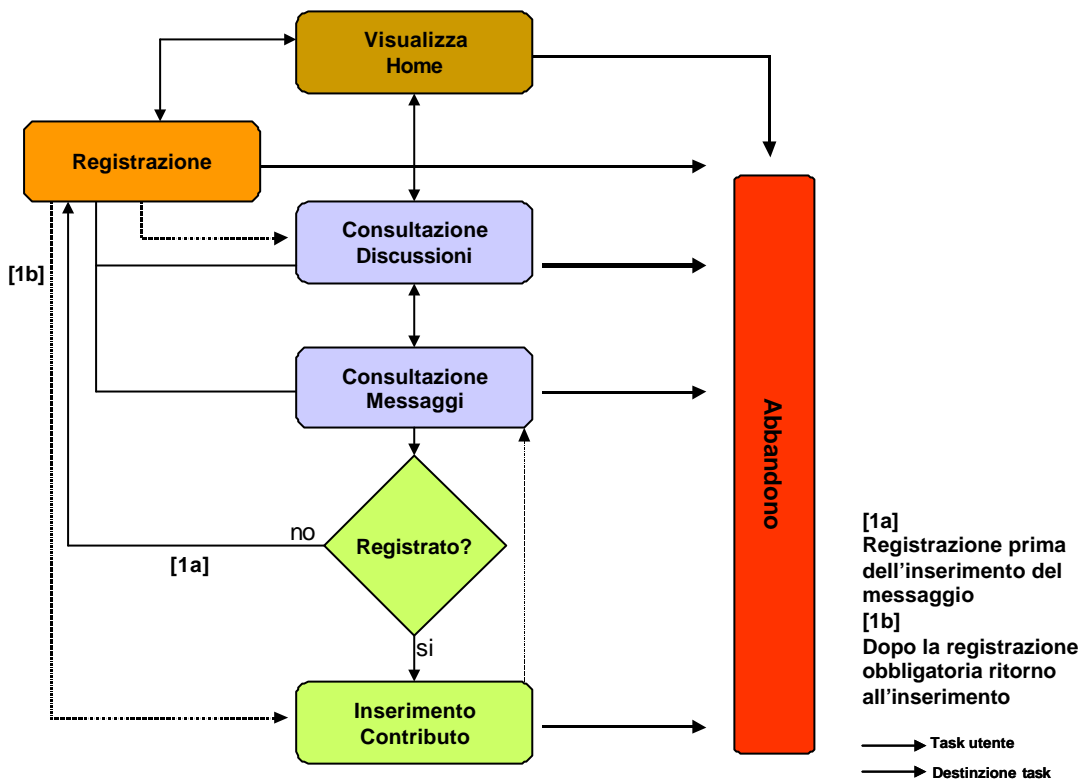
La differenziazione tra gli utenti può essere fatta: con la disabilitazione dell'inserimento nell'interfaccia della pagina oppure con un controllo nella fase d'inserimento del messaggio, con conseguente invio alla pagina di registrazione (o un avviso con la possibilità di accedere alla registrazione).

¹⁹ UML, in particolar modo, è un linguaggio di modellazione che permette di migliorare la comprensione della realtà studiata attraverso diverse tipologie di diagrammi che conducono ad una visione globale del sistema e delle sue astrazioni.



Questi diagrammi descrivono il comportamento del sistema o parte di esso attraverso un set di azioni eseguite da uno o più attori. Questo strumento permette di evidenziare le funzionalità che il sistema mette a disposizione dal punto di vista esterno, dell'utente, come una "black box" sigillata e quindi osservarne solo i comportamenti dall'esterno. E' il punto di vista dell'utilizzatore, e di tutto ciò che interagisce con il sistema nell'ambito del suo funzionamento. I casi d'uso sono i modi in cui il sistema può essere utilizzato e nel caso delle applicazioni web ricalcano in buona parte la navigazione del sito.

²⁰ Da notare l'utilizzo del colore per evidenziare task appartenenti allo stesso obiettivo di navigazione come la consultazione degli argomenti e la consultazione dei contributi che appartengono all'obiettivo di consultazione del Forum.

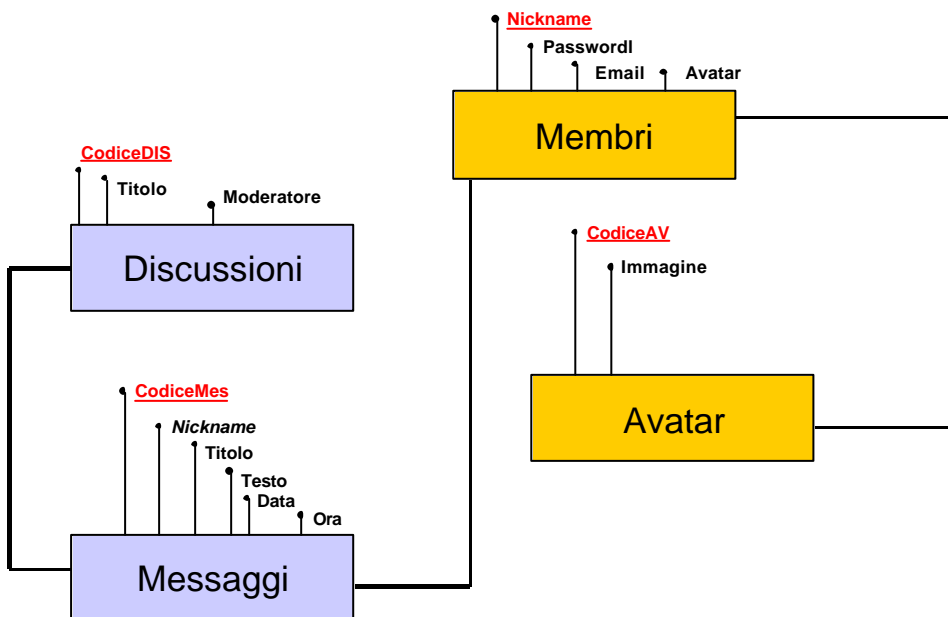


Esempio di task di Alice, Aldo e Anna per un Forum.

Progettazione Database

Il risultato della progettazione è il database relazionale in forma normalizzata.

Nel caso, per esempio, di un Forum organizzato su due livelli il database semplificato potrebbe essere il seguente:



Progettazione sito

Si progetta il sito sulla base delle best practice di usabilità²¹ e linee guida di accessibilità²².

Per esempio, nel caso di un Forum, l'interfaccia della pagina di visualizzazione dei messaggi potrebbe essere la seguente:

Discussione	Usabilità	Numero argomenti	123
Nickname	Titolo	Data	
Farbutto.it12	Quindi...	09/03/2004 17.40	
	Intervento 123		
	<p><i>Plutone ha scritto il 09/03/2004 17.00</i></p> <p><i>Un prodotto usabile fa venire in mente un prodotto che può essere utilizzato in modo facile e amichevole per l'utente, senza la necessità di un lungo e difficile apprendimento, attraverso la lettura di manuali e ripetuti tentativi di utilizzo/apprendimento.</i></p> <p>Un prodotto usabile quindi dovrebbe permettere all'utilizzatore di concentrarsi sul cosa fare e non sul come... esatto?</p>		
	Rispondi	Quote	Profilo
Nickname	Titolo	Data	
Plutone	Usabilità	09/03/2004 17.00	
	Intervento 122		
	<p>Un prodotto usabile fa venire in mente un prodotto che può essere utilizzato in modo facile e amichevole per l'utente, senza la necessità di un lungo e difficile apprendimento, attraverso la lettura di manuali e ripetuti tentativi di utilizzo/apprendimento.</p>		
	Rispondi	Quote	Profilo

Progettazione Script pagina

In questa fase è fondamentale superare la disconnessione tra le pagine mediante dei meccanismi e degli strumenti di passaggio dei parametri tra le pagine, dove per parametro intendiamo una variabile ed un valore come per esempio una password inviata mediante un Form

²¹ La tematica della usabilità è approfondita dettagliatamente nella dispensa "Usabilità dei siti" di Nicola Ceccon (2004) disponibile nello spazio materialestudenti.itiseuganeo.it, in questa dispensa per usabilità si intende la proprietà di un sito di essere facile da utilizzare riducendo al minimo gli sforzi dell'utente per trovare le informazioni attraverso una interfaccia che aiuti invece di ostacolare.

²² La tematica della accessibilità si può approfondire in Internet, in questa dispensa per accessibilità si intende la caratteristica del sito di essere accessibile anche alle persone non normodotate come i non vedenti, per esempio.

Meccanismi per il passaggio dei parametri

Le pagine dell'applicazione interagiscono tra di loro passandosi, con diverse tecniche, dei parametri che determinano la creazione della pagina HTML inviata al browser.

Queste tecniche devono permettere la creazione o invio dei parametri e la lettura o recupero degli stessi.

Esistono diversi tipi di parametri e a seconda del tipo si utilizzano meccanismi diversi:

- Parametri a livelli di pagina.
- Parametri a livello di sessione utente.
- Parametri a livello di applicazione.

Parametri a livelli di pagina

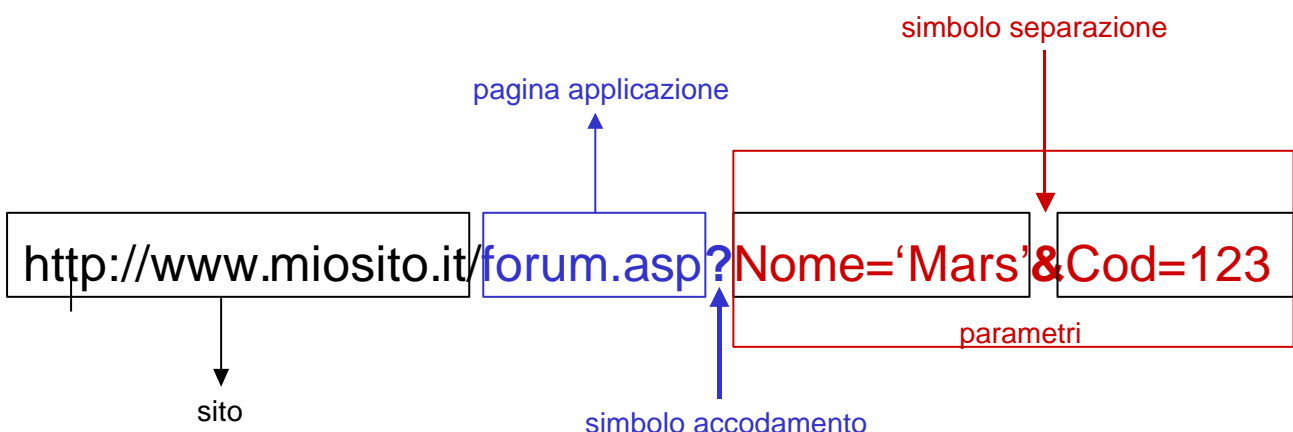
Parametri che devono essere passati da una pagina sorgente ad un'altra di destinazione mediante un form oppure attraverso un collegamento (URL) e che sono **disponibili esclusivamente per la pagina di destinazione** e non per altre.

Distinguiamo tra:

- Parametri form, passati utilizzando il form.
- Parametri URL, passati utilizzando un collegamento²³.

I parametri a livello di pagina sono passati come stringa di testo **accodata**, automaticamente nel caso di form, all'indirizzo della pagina di destinazione in base alla seguente sintassi:

- Il carattere ? separa il nome della pagina dai parametri.
- Coppia parametro e valore separate dal simbolo di =.
- Il simbolo & per separare le coppie.



²³ Definito passaggio mediante URL.

Parametri a livello di sessione utente

Sono parametri accessibili per tutte le pagine, ma esclusivamente al singolo utente durante la sessione²⁴: due o più utenti che accedono alla stessa pagina utilizzano gli stessi nomi di parametri, ma con valori diversi.

La sessione ci permette di stabilire un "dialogo" con l'utente del sito web, superando uno dei limiti del protocollo HTTP, infatti ogni richiesta è unica, e una volta soddisfatta non è più in grado di ricordarla e quindi, quando lo stesso utente presenterà una nuova richiesta, il server web non sarà in grado di rendersi conto che si tratta dello stesso utente della richiesta precedente, e la tratterà quindi come tutte le altre.

Ovviamente, non sempre questo è un problema: la normale navigazione sui siti internet può tranquillamente avvenire in questo modo. Ma ci sono, invece, delle situazioni in cui è importante stabilire questa specie di dialogo tra l'utente ed il server, in quanto ogni operazione che l'utente compie può influenzare le successive ed essere influenzata dalle precedenti.

Un esempio è quello dei siti con aree riservate: se un certo numero di pagine è visibile solo agli utenti registrati, questi dovranno autenticarsi²⁵ per poterle vedere, ma una volta effettuata l'autenticazione dovranno essere messi in grado di vedere tutte le pagine dell'area riservata, senza, ovviamente, doversi autenticare di nuovo per ciascuna pagina: questo sarà possibile solo se il server ricorda che quell'utente si è già autenticato.

I parametri a livello di sessione utente nella programmazione sono utilizzati come variabili globali temporanee o permanenti.

Distinguiamo tra:

- **Parametri di Sessione** passati mediante un apposito oggetto o tecnica del linguaggio del programmazione.
- **Parametri passati mediante Cookies.**

Parametri di Sessione

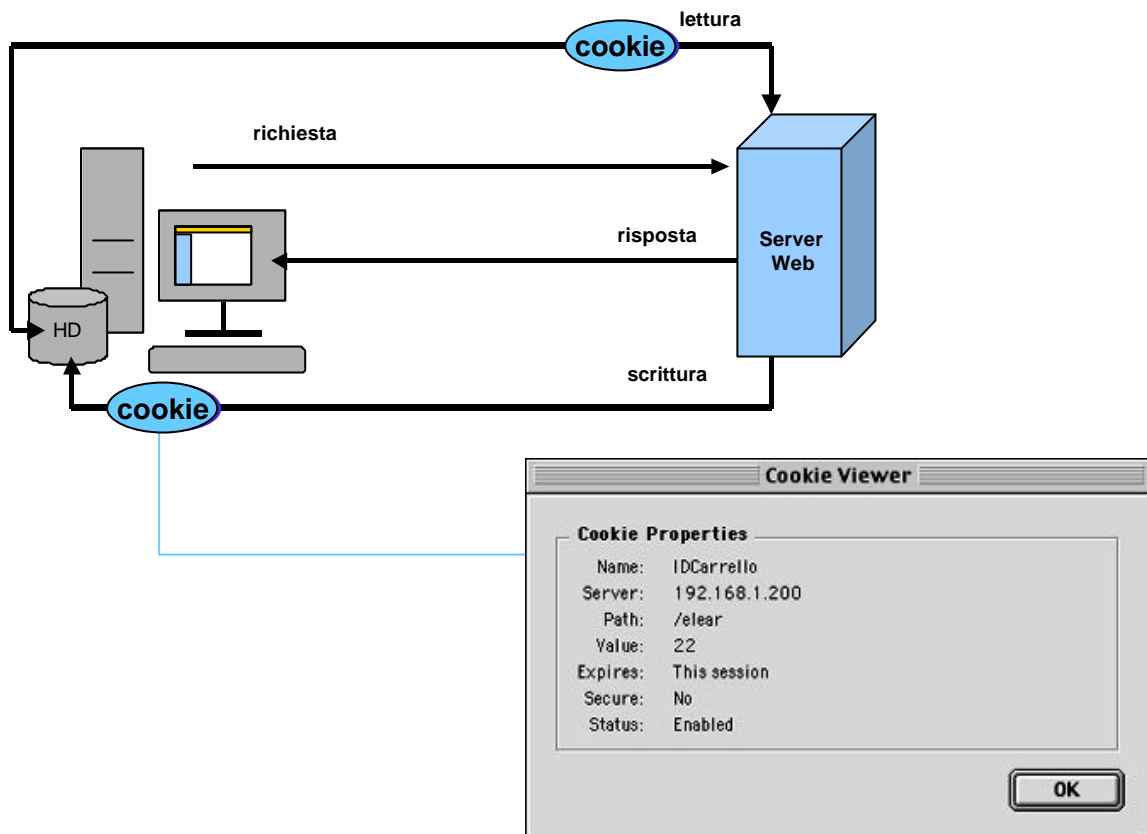
- Sono parametri temporanei la cui durata si può cambiare rispetto a quella di default oltre la quale il sistema li cancella.
- I parametri sono cancellati quando l'utente chiude il browser.
- L'implementazione dipende dal linguaggio di scripting utilizzato.

Cookies

I cookies sono dei file di testo che sono generati e letti sul lato server sul quale è posto il sito web, e che vengono creati e memorizzati sul lato client ottimizzando in tal modo le risorse di memoria sul server.

²⁴ Se i parametri o alcuni devono essere passati dalla pagina di destinazione ad un'altra una tecnica prevede il passaggio mediante form nascondendo il parametro specificando l'elemento del form come HIDDEN (nascosto). Un esempio può essere la necessità di passare da un catalogo dei prodotti alla pagina dettaglio dei prodotti il codice dell'articolo e successivamente da questa alla pagina di inserimento nel carrello della spesa.

²⁵ Procedura di Login.



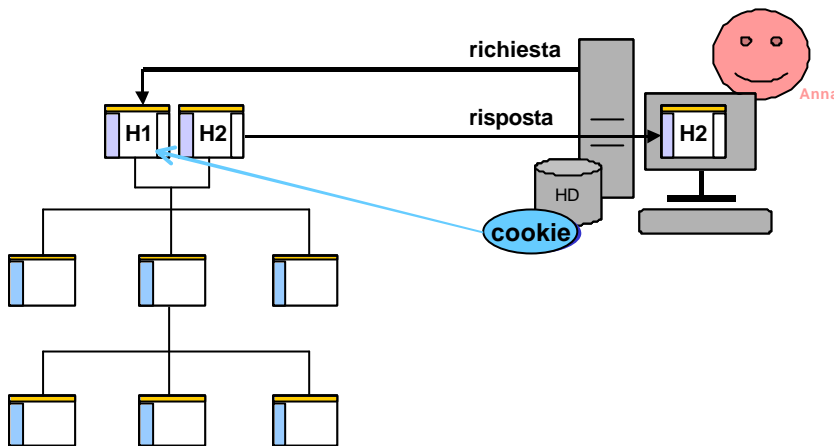
I cookies hanno diverse proprietà tra cui spiccano il nome, il valore, il server e la scadenza.

Esistono due diversi tipi di cookies:

- **Temporanei** che durano sino a che l'utente non chiude il browser e che sono equivalenti ad un parametro Sessione.
- **Permanenti** che durano sino ad una **data di scadenza** impostata indipendentemente dalla chiusura del browser.

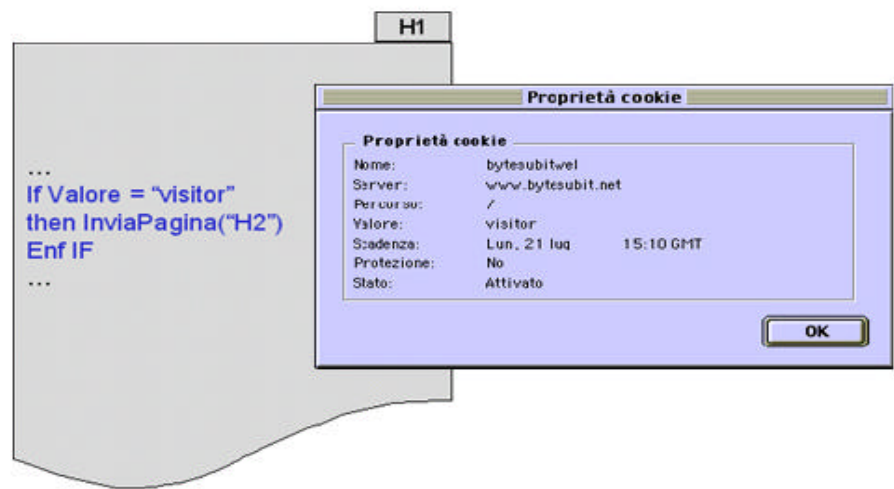
Poiché sono memorizzati sul disco fisso del client, basta solo interrogare i cookies come se fossero semplici variabili, e ciò è possibile farlo anche a distanza di tempo. Infatti un cookies é molto di più di una variabile: una variabile una volta chiusa l'applicazione web, muore con essa, mentre, al contrario, i cookies restano sul computer del visitatore²⁶.

²⁶ Si deve fare attenzione all'utilizzo dei cookies perché l'utente mediamente non li sopporta e tende ad eliminarli oppure a disabilitarne la ricezione. A dire il vero se ne conosce l'esistenza e se è in grado di farlo (cosa del tutto non ovvia).



Per esempio Anna entra nel sito e riceve l'Home page "H2", riservata ai visitatori, che tornano invece dell'"H1" per i visitatori che accedono per la prima volta. Anna è stata riconosciuta dal server web grazie ad un cookie presente nel suo computer in precedenza invitato alla prima visita.

E' soprattutto quella dei cookies una strada molto utilizzata per condividere i parametri tra tutte le sessioni di un utente.



Il cookie, quindi, viene utilizzato quando vi è l'incombenza di memorizzare dei dati in particolari contesti²⁷, un esempio è il carrello in un sito di ecommerce: il codice unico del carrello per la gestione della prenotazione dei prodotti è assegnato all'utente all'accesso della Home page e disponibile in tutte le pagine dell'applicazione. Pertanto, la pagina di inserimento nel carrello effettua l'operazione prelevando (leggendo) il cookie assegnato.

Osserveremo in seguito che al cookie si può far svolgere (creando una specie di emulazione) quasi tutti i tipi di variabili (locali, globali, di sessione). L'unica variabile che non è possibile emulare tramite i cookies è la variabile a livello di applicazione in quanto questa è sfruttata da tutti gli utenti contemporaneamente connessi nel web in un dato istante.

²⁷ Uno svantaggio nell'utilizzo si ha nel caso di dati personali e/o password in quanto i cookies sono dei semplici file di testo e quindi di facile lettura da tutti.

Parametri a livello di Applicazione

Parametri accessibili in tutte le sessioni, quindi per tutti gli utenti: due o più utenti che accedono alla stessa pagina utilizzano gli stessi parametri con gli stessi valori. Un esempio può essere il numero degli utenti collegati in un dato momento oppure una parte del contenuto della Home page dinamica ma comune a tutti gli utenti come le News personalizzate del giorno.

In questo caso si utilizzano gli oggetti e le variabili messe a disposizione dalla tecnologia utilizzata per la realizzazione di applicazioni web.

Visibilità dei parametri

I parametri hanno una diversa visibilità:

- **Parametri a livelli di pagina** sono visibili solo alla pagina di destinazione e solo all'utente coinvolto nella interazione.
- **Parametri a livello di sessione utente** sono visibili in tutte le pagine e solo all'utente coinvolto nella interazione.
- **Parametri a livello di applicazione** sono visibili in tutte le pagine e per tutti gli utenti coinvolti nella interazione.

Esempio chiarificatore di utilizzo dei parametri

Supponiamo che un navigatore accedi alla Home page di un Forum.

1. Riceve la pagina con il numero degli utenti collegati al momento del trasferimento, funzione svolta dall'incremento del contatore dei navigatori collegati, reso disponibile a tutte le sessioni come parametro di Applicazione.
2. Se il navigatore è riconosciuto come utente registrato il suo Nickname viene reso disponibile per tutte le pagine dell'applicazione come parametro di Sessione, a seguito della lettura di un cookie permanente contenente il codice di registrazione e di una interrogazione al database per ricavarne il Nickname.

Se non è riconosciuto gli viene assegnato il Nickname generico "Ospite" che non permette l'inserimento, ma solo la lettura dei contributi.

L'utente ha quindi innescato la creazione dei parametri di Applicazione e di Sessione utente al semplice accesso alla Home page senza che abbia manifestato in modo esplicito la volontà di partecipare al Forum.

3. Dopo la navigazione a partire dagli argomenti trova la discussione che gli interessa e scrive un contributo accompagnato dalla propria Password che invia con un form, generando dei parametri Form a livello di pagina.
4. Al comando di invio è associato l'elaborazione di una pagina di inserimento che riceve i parametri di tipo Form e per prima cosa si occupa del controllo della Password.

41. **Nel caso di matching** il codice effettua l'inserimento nella tabella del database che memorizza i contributi un record costituito:

- Dal codice dell'argomento ottenuto come parametro di pagina di tipo Form.
- Dal codice dell'utente ottenuto come Cookie.
- Dalla data ottenuta dal sistema come funzione del linguaggio di scripting client utilizzato.
- Il testo dell'argomento ottenuto come parametro Form.

La pagina della applicazione web oltre all'operazione di inserimento crea la risposta per l'utente facendo visualizzare i contributi dell'argomento in corso, cioè invia all'utente la pagina di visualizzazione dei contenuti.

42. **Nel caso di fallimento**, cioè caso di utente non registrato, si invia una pagina di risposta con la possibilità di collegarsi alla pagina di registrazione²⁸.

Parte di questo codice HTML viene codificato esplicitamente all'interno della pagina dallo sviluppatore, mentre la parte restante viene creata dallo script utilizzato per l'inserimento.

5. Il navigatore continua a navigare ed esce dal sito.

Quando la sessione è completata, dall'utente (chiusura del browser) o dal server web (tempo scaduto) tutti i parametri sono azzerati, tranne, però, quelli di Applicazione.

Schematizzazione

La progettazione produce degli schemi, cioè una schematizzazione dell'applicazione. Allo scopo risulta essere molto comodo oltre agli schemi dei task e del database produrre:

1. Schema della rappresentazione del sito o della parte dell'applicazione web in termini di **relazioni tra pagine**.
2. Schemi di ogni pagina dell'applicazione con lo **pseudoscript delle pagine**.

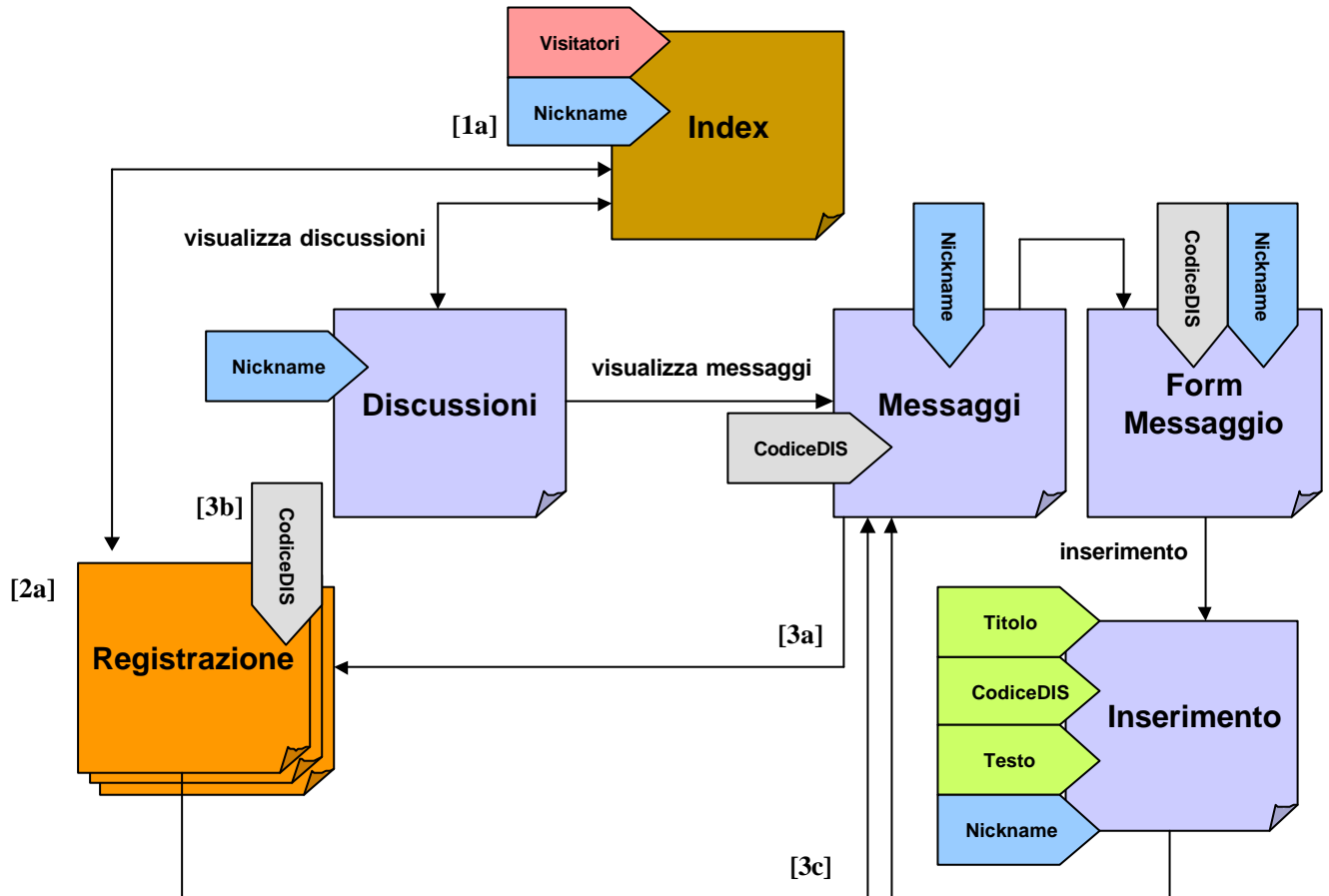
Relazione tra le pagine

Partendo dai task individuati, si procede alla schematizzazione dell'applicazione individuando le pagine, considerando che un task può essere assolto da più pagine e viceversa una pagina può assolvere diversi task, e al contempo evidenziando la relazione tra le pagine e i diversi tipi di parametri utilizzati.

²⁸ Una soluzione preferibile è quella di presentare nella pagina di risposta il form di registrazione.

Lo schema viene espresso con diagrammi²⁹, eventualmente, accompagnato da note e specifiche. In questo diagramma si evidenzia il parametro che la pagina riceve o genera.

L'esempio che si presenta è relativo ad una schematizzazione parziale di una applicazione per la realizzazione di un Forum che evidenzia le operazioni di ricerca della discussione, ricerca dei messaggi e l'inserimento di un messaggio (all'interno della discussione che si sta visualizzando).

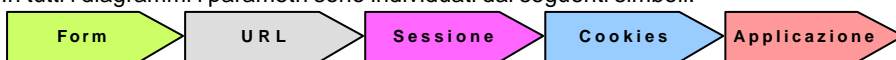


[1a]
Se l'utente non dispone del Cookie Nickname perché non registrato, viene creato il Cookie con nome Ospite, che quindi identifica nel sistema un utente non registrato, altrimenti viene identificato.

[2a]
La funzione di registrazione è rappresentata come gruppo di pagine rappresentante il form di registrazione, il controllo e la gestione dell'errore.

[3a]
Inserimento di un messaggio da parte di un utente non registrato (Nickname Ospite).

²⁹ I diagrammi non appartengono ad uno standard, ma sono stati creati appositamente per chiarire le relazioni tra le pagine e successivamente il pseudocodice di scripting. In tutti i diagrammi i parametri sono individuati dai seguenti simboli:



[3b]

Il parametro che identifica la discussione, CodiceDIS, viene passato come parametro URL alla pagina di registrazione dalla pagina messaggi e successivamente come parametro URL alla pagina messaggi per poter visualizzare la discussione in cui si trovava.

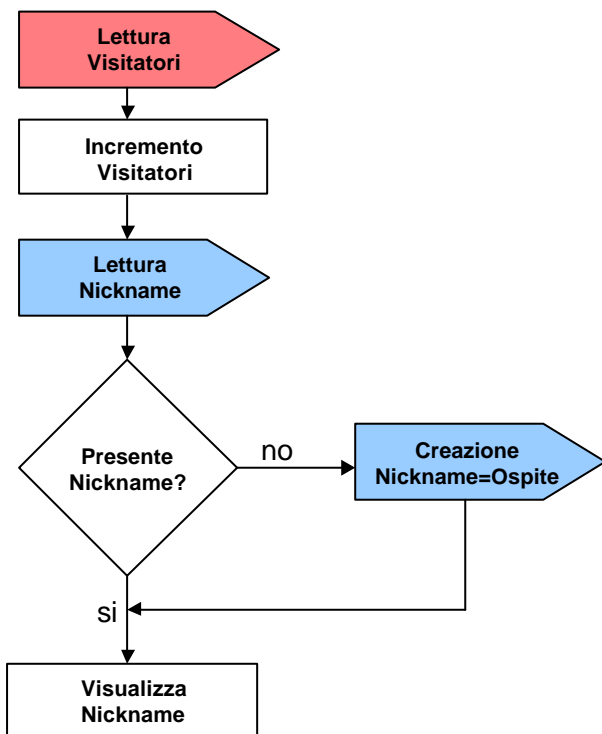
[3c]

Ritorno alla pagina dei messaggi della discussione scelta da Ospite prima della registrazione, in questo modo l'utente effettua l'inserimento dal punto che aveva generato la richiesta.

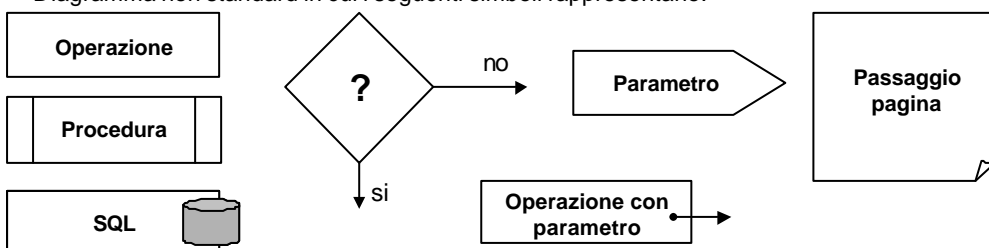
Pseudoscript delle pagine

In questa fase si evidenzia il codice della pagina dinamica a livello di pseudocodice utilizzando dei diagrammi³⁰ a blocchi come negli esempi chiarificatori che seguono.

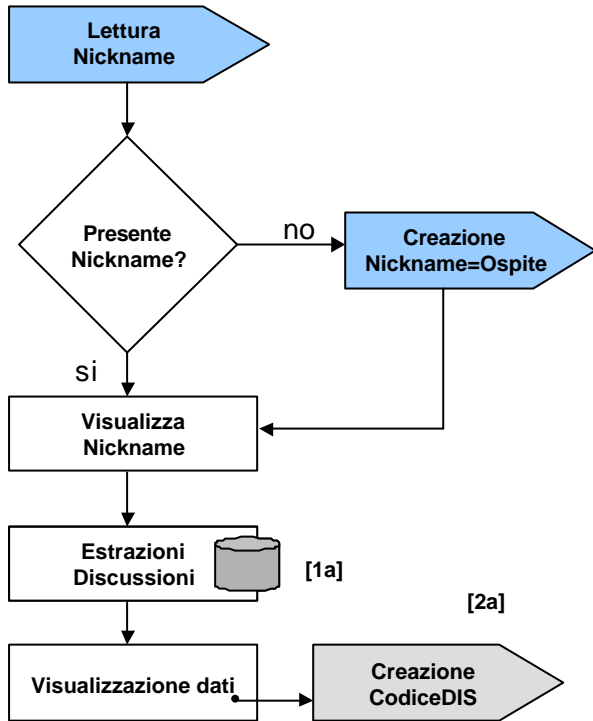
Home page



³⁰ Diagramma non standard in cui i seguenti simboli rappresentano:



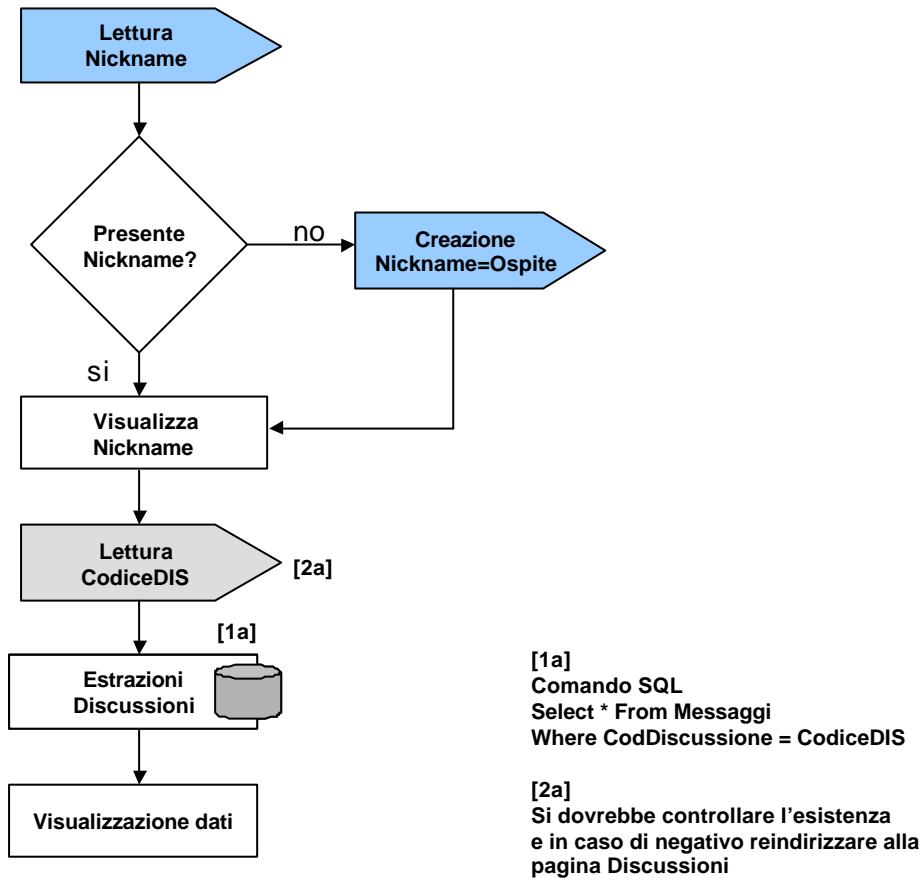
Visualizza discussioni



[1a]
Comando SQL
Select * From Discussioni

[2a]
Creazione del collegamento ai
messaggi di ogni discussione come
parametro URL nome CodiceDIS

Visualizza messaggi



Tutorial: applicazione di ecommerce

Organizzazione del sito

Il sito di ecommerce è organizzato:

- Sul **processo di vendita-acquisto**, che costituisce il cuore dell'applicazione:
 1. Ricerca del prodotto mediante la consultazione del catalogo.
 2. Analisi del prodotto individuato.
 3. Eventuali richieste di consigli.
 4. Acquisto del prodotto.
 5. Consegna del prodotto.
- Su una serie di **servizi aggiuntivi**³¹ che sfruttano le caratteristiche del web per aumentare le vendite: recensione di un prodotto, consiglio di utilizzo, "Hot product" (novità, prodotti in offerta, più venduti), prodotti **similari**³² a quello cercato, mailing list, newsletter, Forum, sezione regalo (possibilità di inserire un messaggio, selezione del biglietto, selezione della confezione³³), sondaggi, giudizi di altri utenti che hanno acquistato e provato il prodotto.

Strategie dell'acquisto online

Ci sono due diverse impostazioni della strategia d'acquisto:

1. Una **versione dbcentrica**, cioè fondata completamente sul database in cui il processo³⁴ di vendita-acquisto è totalmente gestito dall'applicazione web.
Si tratta indubbiamente dell'impostazione più complessa ed è quella che si analizza in questa dispensa.
2. Una versione **parzialmente connessa al database**, con una sincronizzazione dei dati in sede di conferma dell'ordine anche usando metodi tradizionali (Fax) ed eventualmente la ricezione del pagamento (Bonifico bancario o Vaglia postale).

³¹ Non necessari, ma fortemente auspicabili per poter ottenere l'obiettivo di trasformare il visitatore in un cliente diretto, se acquista online, o indiretto, se acquista nel negozio dopo avere assunto informazioni nel sito.

³² Un esempio "illuminante" di proposta di articoli simili è dato da Amazon, in cui la scelta dei libri presentati nella stessa pagina di quello cercato, avviene con il criterio dell'acquisto: si trova, cioè, la dicitura "Chi ha comprato questo libro, ha anche acquistato...", seguita dalla lista dei testi correlati. Il cliente che, per qualche motivo, non sia soddisfatto dal libro di cui ha appena finito di leggere la descrizione o recensione, può allora facilmente passare in rassegna un altro che parli dello stesso argomento o che, comunque, sia stato acquistato dallo stesso "tipo di cliente". Amazon utilizza uno strumento di analisi degli acquisti allo scopo di pervenire a due risultati:

1. Avere una statistica sui vari tipi di libri e sui suoi clienti (quanti clienti di romanzi, quali tipi di romanzi sono i più richiesti...).
2. Proporre ad ogni cliente un percorso personalizzato attraverso gli argomenti richiesti.

³³ Packaging.

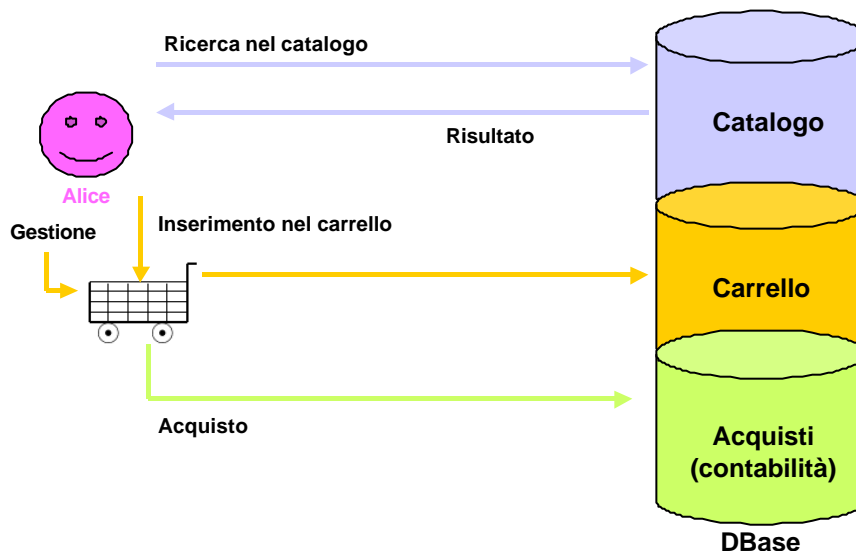
³⁴ Nel caso di vendita di prodotti digitali (musica, software, libri elettronici...) anche la consegna è digitale e quindi il processo è pienamente digitale.

In questo caso l'applicazione web gestiscono la prenotazione dell'acquisto e quindi non l'intero processo di vendita-acquisto.

Versione dbcentrica

In questo caso il carrello della spesa viene tramutato in ordine dall'applicazione web e il database, pertanto, risulta costituito da tre parti:

1. La parte che si occupa della gestione del catalogo, che diventa l'input per il carrello.
2. La parte che si occupa della gestione del carrello.
3. La parte che si occupa dell'acquisto, cioè una volta che il navigatore ha concluso il processo di acquisto il carrello diventa l'input per l'aggiornamento delle tabelle del database che gestiscono normalmente gli acquisti (ordini, clienti, transazioni economiche).



Elementi fondamentali

Un sito di ecommerce è, oramai, basato su due interfacce dominanti³⁵:

1. La **navigazione tra i prodotti** disponibili attraverso una loro suddivisione per categoria, come nei normali cataloghi cartacei.
2. L'utilizzo di un **carrello della spesa** per la scelta dei prodotti e l'invio dell'ordine.

Il carrello contiene i dati fondamentali del prodotto, la quantità scelta, i costi parziali, il costo totale, eventuale IVA, gli eventuali costi di trasporto o riferimento ad un successivo costo di questo tipo nel momento della scelta del tipo di consegna.

³⁵ Sono praticamente degli standard.



Gli elementi fondamentali di un sito di ecommerce sono:

- Gli strumenti di navigazione.
- La gestione del carrello.
- Il percorso verso la cassa.
- La sicurezza della transazione.
- L'interfaccia.

Gli strumenti di navigazione: la navigazione tra gli scaffali del negozio elettronico non viene in modo lineare, cioè non esiste un percorso unico verso il prodotto.

Strumenti di navigazione

Le metodologie di navigazione sono:

- La navigazione per **link proposti**.
- La navigazione per **categorie e sottocategorie** (albero o directory)
- L'utilizzo di un **motore di ricerca**: ricerca immediata (semplice query) e ricerca avanzata (query con filtri). E' fondamentale che il sistema di ricerca sia progettato per intercettare i sinonimi ed anche le ricerche con errori di digitazione, di spelling o con poche informazioni (per esempio "Gibson" o "Gibbson" o "Gibso" nel caso dello scrittore Wiliam Gibson). E' nella ricerca diretta che emerge la grande differenza tra il negozio tradizionale e quello online: il click sul prodotto porta il navigatore (potenziale cliente) esattamente nella pagina del prodotto selezionato, senza ulteriori ricerche e spostamenti "fisici".

La gestione del carrello: il carrello (Cart o shopping cart) della spesa online, ha il suo corrispondente nella realtà: è un contenitore dove si accumulano i prodotti scelti, prima di arrivare alla cassa per pagare.

Gestione carrello

Quindi, il carrello deve essere completamente gestibile dal navigatore:

1. Inserimento di un prodotto, con il link "aggiungi al carrello".
2. Modifica della quantità.
3. Cancellazione di un prodotto.

³⁶ Immagine da www.amazon.com

4. Cancellazione dell'intero carrello.

Il percorso verso la cassa: l'acquisto nel virtuale comporta una serie di passaggi allo scopo di assumere i dati anagrafici, il tipo di consegna e di pagamento (eventualmente e sempre opzionale il numero della carta di credito). La metodologia di assunzione dei dati anagrafici dipende se il navigatore è già registrato, cioè è un cliente, oppure no, cioè al primo acquisto.

Verso la cassa

Nel primo caso i dati si assumono con una procedura di autenticazione³⁷ rapida con l'introduzione di un dato identificativo, come un Nickname oppure un indirizzo di Email valido, e una Password. Questa procedura è spesso identificata come "acquisto con 1-Click".

1-Click

The screenshot shows the 'Sign In' form on Amazon.com. It has a title 'Sign In' in orange. Below it is the question 'What is your e-mail address?' followed by a text input field. The second question is 'Do you have an Amazon.com password?' with two radio button options: 'No, I am a new customer.' and 'Yes, I have a password:' followed by a password input field. At the bottom, there is a yellow button with a right-pointing arrow and the text 'Sign In using our secure server'. The number '38' is in the bottom right corner of the form's bounding box.

Nel secondo caso, il primo acquisto, i dati sono assunti con un Form ed il sistema riceve e fornisce un nome identificativo e una password per la futura autenticazione.

The screenshot shows the 'Registration' form on Amazon.com. It has a title 'Registration' in orange. Below it is the question 'New to Amazon.com? Register Below.' followed by several input fields: 'My name is:', 'My e-mail address:', and 'Type it again:'. There is also a 'Birthday:' section with dropdown menus for 'Month' and 'Day' and the text '(optional)'. Below this is another section titled 'Protect your information with a password' with the subtext 'This will be your only Amazon.com password.' followed by two password input fields: 'Enter a new password:' and 'Type it again:'. At the bottom, there is a yellow button with a right-pointing arrow and the text 'Continue'. The number '39' is in the bottom right corner of the form's bounding box.

La sicurezza della transazione: cioè il pagamento, è l'elemento più "delicato" perché è l'ultimo ostacolo all'acquisto.

E' fondamentale fornire sicurezza e fiducia attraverso una informazione chiara

³⁷ Detta anche di Login o Sign Up.

³⁸ Immagine da www.amazon.com

³⁹ Immagine da www.amazon.com

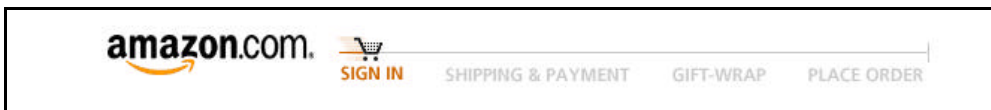
e puntuale, l'appoggio a servizi bancari noti e utilizzo di certificazioni esterne⁴⁰ di sicurezza e affidabilità.

L'interfaccia: il successo di un sito di ecommerce è strettamente legato all'interfaccia, cioè alla facilità di comprensione e utilizzo dei diversi elementi del sito.

Interfaccia

In particolare risulta utile, se non indispensabile, fornire l'informazione puntuale sul percorso verso la cassa, cioè l'utente deve chiaramente capire a che punto si trova e quanto manca all'arrivo (oltre a sapere dove sta andando).

In molti casi si utilizza un elemento di interfaccia come nella figura⁴¹ che segue:



Una progettazione carente di uno di questi elementi comporta il non acquisto, come molti studi dimostrano, e lo scopo dei progettisti è quello di creare un ambiente facile, confortevole, credibile e sicuro.

Errori fondamentali

Non sempre, però, il sito risponde allo scopo per cui è stato creato, cioè permettere di creare uno scambio di valore, a causa di una serie di errori di progettazione.

I principali errori⁴² sono:

- **Occultazione del prodotto:** nella Home page si **focalizza la comunicazione sull'azienda e non sul prodotto**, cioè il catalogo online è disponibile come collegamento all'interno degli altri collegamenti relativi all'azienda con il rischio che si perda e senza evidenziare immediatamente che il catalogo è strutturato in categorie. E', invece, opportuno evidenziare nella Home page la struttura del catalogo permettendo l'accesso diretto alle categorie ed eventualmente sottocategorie.
- **Disponibilità solamente del catalogo off-line** da scaricare: il catalogo dovrebbe essere sempre disponibile per la navigazione online, eventualmente con una versione alternativa per lo scaricamento.
- **Non mostrare alcuna immagine del prodotto:** spesso, invece, una immagine comunica più del testo.



⁴⁰ Per esempio la rivista dei consumatori "Altroconsumo" fornisce un bollino "Webtrader" ai siti (aziende) che aderiscono ad un sistema di certificazione basato su un codice di buona condotta relativo a correttezza e completezza informazioni, sicurezza pagamenti, diritto di recesso.

⁴¹ Immagina da www.amazon.com

⁴² Evidenziati da Sofia Postai in un articolo su Internet News del settembre 2003.

- **Gallerie di immagini senza contesto e senza dettagli:** le immagini troppo piccole non forniscono informazioni e nel caso non ci sia spazio nella pagina sufficiente è preferibile rimpicciolire un dettaglio che renda comunque riconoscibile il prodotto piuttosto di tutta l'immagine.



- **Mancanza di pagine di approfondimento dei prodotti:** la possibilità di disporre di una pagina di approfondimento è indispensabile poiché l'utente deve avere le informazioni per effettuare le sue valutazioni (anche attraverso una funzione di comparazione tra i prodotti, dato che in un negozio un utente entra in relazione diretta con il prodotto).
- **Mancata evidenziazione delle offerte e delle promozioni:** queste permettono di attirare l'utente contribuendo alla navigazione su alcuni settori, anche profondi, di produzione.

Come Postai⁴³ evidenzia:

Il punto focale di ogni sito aziendale deve essere quello di presentare al cliente quello che al cliente interessa. E' fondamentale resistere alla tentazione di usare il sito aziendale come un'occasione per esibire foto dei dirigenti, dei propri capannoni, dei premi, della propria storia, a discapito del prodotto. Un sito dove l'azienda si pavoneggia su se stessa o coglie solo l'occasione per sfoggiare immagini attraenti ma fuori tema è un'occasione sprecata che genera disaffezione e a volte irritazione da parte dell'utente. Un'azienda interessa se può offrire un buon prodotto/servizio all'utente. E' dunque il prodotto a dover essere messo costantemente in primo piano, studiando anche il modo in cui l'utente interessato acquisisce progressivamente informazioni e forma le sue decisioni di acquisto.

- **Ambiguità dei costi:** nella visualizzazione del carrello devono esserci i costi parziali e il costo totale. Deve anche essere specificato che i costi di trasporto non sono calcolati, dato che dipendono dal tipo di consegna. Inoltre si deve specificare l'aspetto dell'IVA (esente o presente).

⁴³ Sofia Postai in un articolo su Internet News del settembre 2003.

Tutorial: il progetto dell'applicazione ecommerce

Premessa

Nel tutorial si presenta e si analizza una soluzione di ecommerce⁴⁴ per l'ipotetica azienda "VestiGiusto" che produce e commercializza esclusivamente online articoli e prodotti di abbigliamento.

In dettaglio l'applicazione, quindi il sito, deve permettere:

- La consultazione del catalogo dei prodotti.
- Ricerca di un prodotto con un motore di ricerca interno
- Il deposito dei prodotti di suo interesse nel carrello della spesa (basket o shopping cart) in un qualsiasi momento senza necessità di registrazioni o altre procedure esplicite di identificazione.
- Conferma dell'acquisto nella fase di Chek Out.
- Scelta del corriere e tempi di consegna.
- La Registrazione allo scopo di diventare un utente registrato, conosciuto al sito, per poter effettuare l'acquisto più velocemente.

In questo caso l'utente ha già inserito i dati necessari (e sensibili) e l'acquisto consiste nella introduzione del nominativo e password di accesso scelta dall'utente nella registrazione.

Visualizzazione dei prodotti in promozione nella Home page

- L'iscrizione alla Mailing List delle Only you (News), che comporta automaticamente anche la Registrazione.
- Il tracciamento degli ordini effettuati: il cliente inserendo il codice identificativo dell'ordine nel sito dello spedizioniere può controllare lo stato del proprio ordine (in magazzino, in viaggio, imminente consegna).
- Sincronizzazione automatica tra la richiesta di acquisto e la gestione di un ordine.

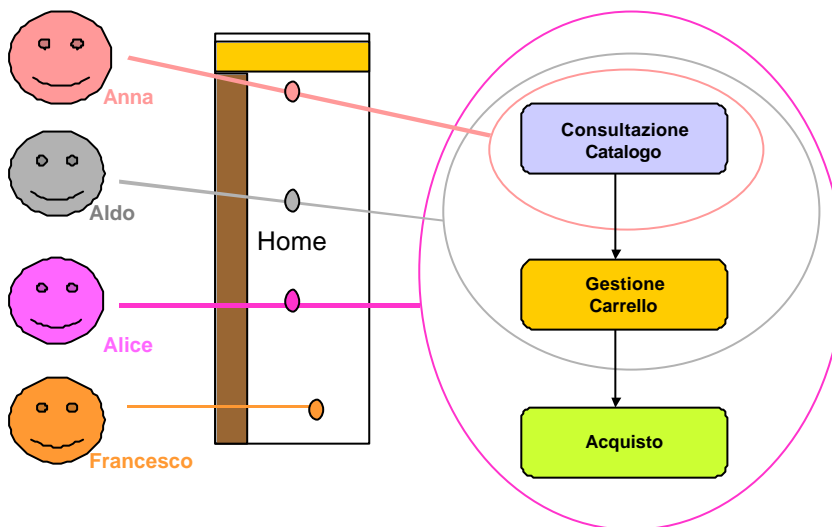
⁴⁴ Parte delle funzionalità dell'applicazione sono disponibili come pagine in Microsoft .

Business Logic

Scenari

Nel procedere ad una accurata progettazione dell'applicazione per assolvere al meglio il processo di vendita-acquisto di un utente, risulta fondamentale partire dalla individuazione degli scenari di navigazione in modo da rappresentare tutte le possibili interazioni.

Nel caso di un sito per la vendita di capi e accessori di abbigliamento sportivi si possono individuare 4 tipologie di utenti:



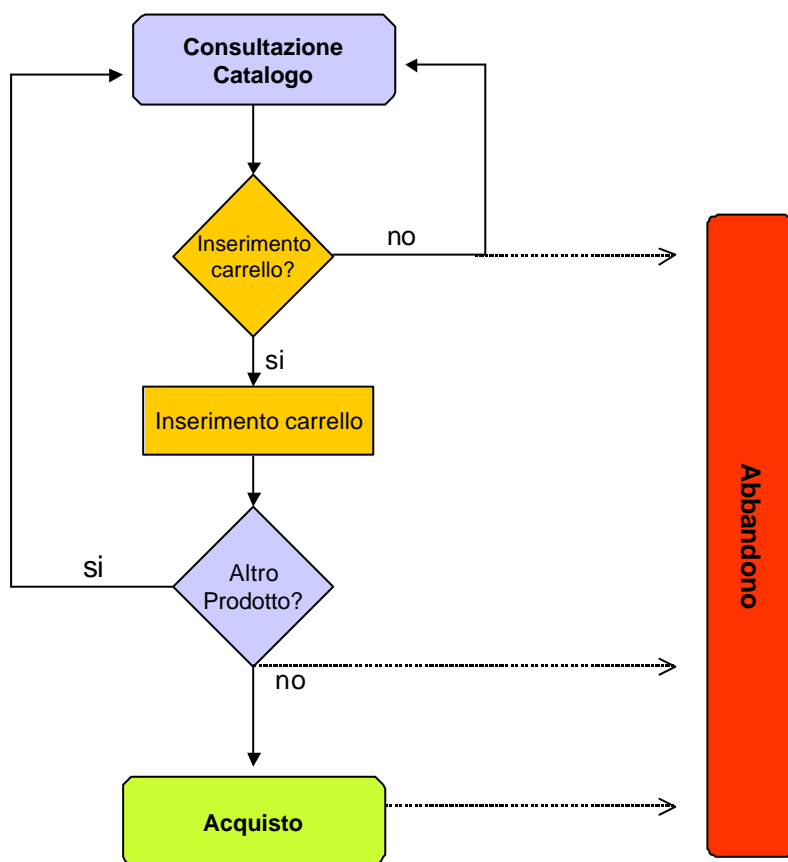
- **Anna**, studentessa universitaria di 22 anni, ama l'abbigliamento sportivo e utilizza il web per cercare questo tipo di abbigliamento per poi, eventualmente, acquistare in negozio. Non ha mai effettuato un acquisto online. Utente che non utilizza il carrello.
- **Aldo**, tecnico di 39 anni, veste anche sportivo ed è molto attento al prezzo prima di effettuare un acquisto. E' sua abitudine una volta individuato il prodotto che lo interessa confrontare il prezzo dei diversi venditori. Ha già effettuato acquisti online, ma sempre in contrassegno per dubbi sulla sicurezza. Particolarmente attento alla Privacy è sua abitudine disabilitare la ricezione di ActiveX, Cookies e codice di scripting JavaScript. Utilizza il carrello per depositare i prodotti che ha trovato interessanti per poi effettuare una stampa anche a scopi comparativi.
- **Alice**, segretaria di 34 anni, ama l'abbigliamento casual ricercato ed è una vera patita dello shopping anche online. Ha già effettuato vari acquisti online e sempre con l'utilizzo della carta di credito.
- **Francesco**, operaio di 44 anni capitato a seguito di una ricerca nel web alla Home page, abbandona il sito perché non è interessato alla tipologia di prodotti proposti.

Progettazione dei task

Il processo si suddivide in tre sottoprocessi:

1. Consultazione del catalogo.
2. Inserimento nel carrello.
3. Acquisto.

L'utilizzo del carrello fa emergere come la consultazione del catalogo sia un processo di ricerca-scelta dell'articolo in cui la scelta comporta l'inserimento dell'articolo nel carrello.



Consultazione del catalogo

Questa parte del processo anche se a prima vista sembra di facile progettazione/implementazione nasconde diverse insidie ed è fonte di errori di usabilità con il conseguente probabile fallimento⁴⁵ del sito.

È prioritario progettare una interfaccia e metodologie di navigazione che permettano velocemente di assumere informazioni sui prodotti.

A questo livello risulta indispensabile permettere l'accesso ai prodotti nella

⁴⁵ Il sito non risponde alla richiesta di informazioni dell'utente.

forma della directory (o albero), similmente alle “pagine gialle”, utilizzando un approccio per categorie.

La struttura del catalogo si organizza in categorie incapsulate⁴⁶, si parte cioè dalle categorie generali per arrivare al dettaglio del prodotto attraverso diversi livelli di sottocategorie (e quindi diversi Click).

In questa organizzazione ogni categoria ha un padre tranne quella di primo livello che non ammette padre: dal punto di vista operativo si utilizza una tabella delle categorie con l'introduzione, oltre al codice della categoria e alla descrizione, l'attributo o campo “categoria-padre”, che serve proprio per organizzare in modo gerarchico la struttura creando delle relazioni padre-figlio tra i record. Infatti, i valori del campo categoria-padre sono i codici delle categorie e si identifica la categoria di primo livello con un valore di codice non presente, come lo zero. Tutti i record padre sono quelli visualizzati nella pagina iniziale del catalogo.

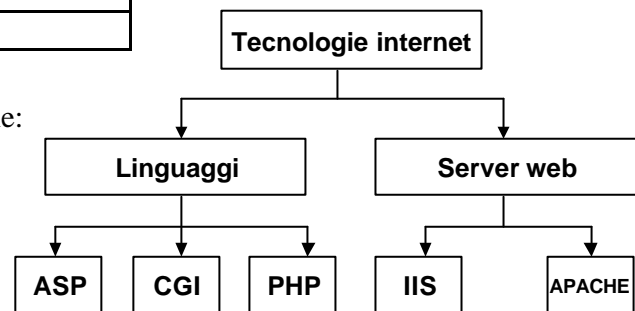
Padre-figlio

Quindi, per esempio nel caso di un database per una libreria online, la tabella delle categorie relativa alle “Tecnologie internet” potrebbe essere:

CodiceCAT	Categoria-Padre	Descrizione
1	0	Tecnologie internet
2	1	Linguaggi
3	2	ASP
4	2	CGI
5	2	PHP
6	1	Server web
7	6	IIS
8	6	Apache

La struttura comporta la seguente lista di categorie:

- Tecnologie internet.
- Tecnologie internet > Linguaggi
- Tecnologie internet > Linguaggi > ASP
- Tecnologie internet > Linguaggi > CGI
- Tecnologie internet > Linguaggi > PHP
- Tecnologie internet > Server web
- Tecnologie internet > Server web > IIS
- Tecnologie internet > Server web > Apache



Il primo errore da evitare è quello di abusare con il numero dei livelli⁴⁷

⁴⁶ Si pensi ad un sito per la vendita di prodotti per l'informatica organizzato su tre livelli: le categorie al primo (Pc di marca, stampanti, monitor...), i prodotti di ogni categoria al secondo (monitor Flatron LG, monitor Flatron LCD...) e il dettaglio al terzo livello (dettaglio del monitor Flatron LG).

⁴⁷ Il numero dei livelli dipende anche dal sito. Per esempio nel caso di un sito per la vendita di

costringendo il navigatore ad un elevato numero di attivazioni di collegamenti, detti click: in generale si considera ottimale un sito che non superi i 3 click.

Il tutorial

Nel caso del nostro tutorial il catalogo è basato su due livelli⁴⁸: cioè l'articolo di abbigliamento al primo e le sue diverse configurazioni al secondo, quindi per esempio all'articolo "Sacco a pelo" sono associate le caratteristiche distinte per forma/taglia, colore... e la disponibilità in magazzino.

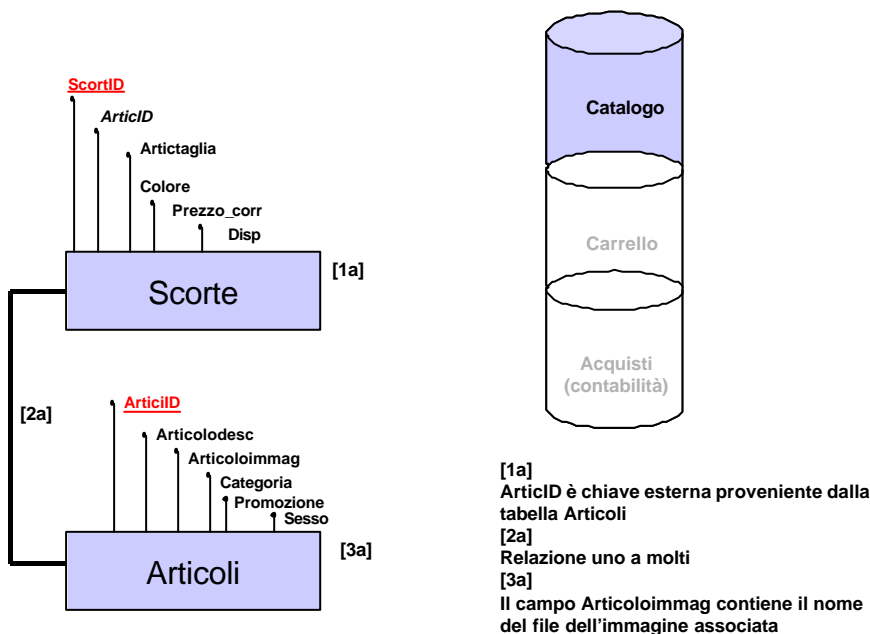
La figura che segue ne rappresenta un esempio:



Codice	5
Descrizione	Sacco a pelo
Categoria	Tempo libero
Sesso	Unisex

Taglia	Colore	Prezzo	Disponibilità
Rettangolare	Azzurro	260,00 €	16
Sagomato	Azzurro	360,00 €	12

Il catalogo ruota attorno alla parte del database utilizzata per memorizzare e organizzare gli articoli disponibili.



prodotti elettronici sicuramente c'è un livello in più rispetto al sito di vendita per prodotti di informatica (la categoria del primo livello diventa la tipologia di prodotto: informatica, elettrodomestici, televisori, hifi...).

⁴⁸ La categoria di appartenenza non è utilizzata per la strutturazione.

Le tabelle possono per esempio contenere i seguenti dati, compatibili con la struttura:

Articoli

ArtID	Articolodesc	Categoria	Sesso	Promozione	Articoimmag
1	Pantaloncini sportivi donna	Pantalone	Donna	N	shorts.jpg
2	Giaccone sportivo donna	Giaccone	Donna	N	fleece.jpg
3	Scarpe da barca	Scarpe	Uomo	N	shoes.jpg
4	Parka	Giaccone	Uomo	S	parka.jpg
5	Sacco a pelo	Tempo libero	Unisex	S	bags.jpg

Scorte

ScortID	ArticID	Artitaglia	Colore	Prezzo_corr	Disp
1	5	Sagomato	Azzurro	260,00 €	16
2	5	Rettangolare	Azzurro	360,00 €	12

Si noti che della tabella Scorte si sono evidenziate le configurazioni dell'articolo "Sacco a pelo", allo scopo di fare emergere la relazione tra le tabelle e la struttura del catalogo.

Il catalogo visualizzato dall'utente contiene i dati rilevanti per il cliente e quindi non contiene i codici e altri dati per il funzionamento del sistema.

L'utente, però, visualizza:

Articoli

Descrizione	Categoria	Sesso	Promozione
Pantaloncini sportivi donna	Pantalone	Donna	N
Giaccone sportivo donna	Giaccone	Donna	N
Scarpe da barca	Scarpe	Uomo	N
Parka	Giaccone	Uomo	S
Sacco a pelo	Tempo libero	Unisex	S

Scorte

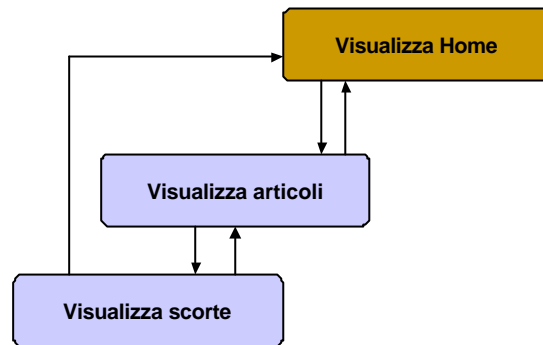
Taglia	Colore	Prezzo	Disponibilità
Rettangolare	Azzurro	260,00 €	16
Sagomato	Azzurro	360,00 €	12

La consultazione del catalogo è un task che comporta la ricerca del capo di abbigliamento di interesse a partire dall'elenco degli articoli per poi controllare se il colore, la taglia e il prezzo sono disponibili.

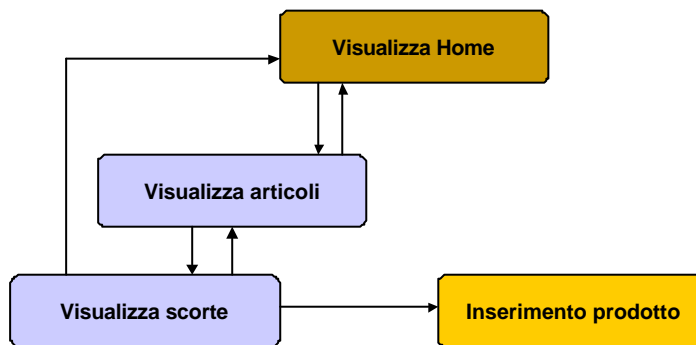
L'utente si può, però, trovare nella condizione di aver individuato l'articolo, ma impossibilitato ad acquistare, causa della indisponibilità in magazzino.

L'operazione di ricerca coinvolge due task elementari:

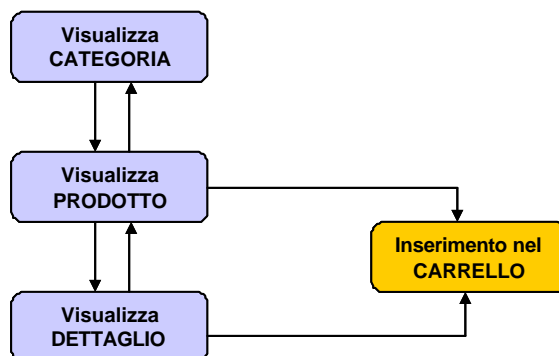
1. La consultazione degli articoli presenti.
2. La consultazione delle scorte con la disponibilità in magazzino.



Nel tutorial l'inserimento è a livello della visualizzazione delle scorte.



In generale in un sito di ecommerce la consultazione del catalogo è finalizzata all'acquisto e quindi si deve rendere disponibile la possibilità di inserire l'articolo nel carrello per l'eventuale acquisto a livello della descrizione del prodotto o del dettaglio se presente (sempre partendo dalle categorie).

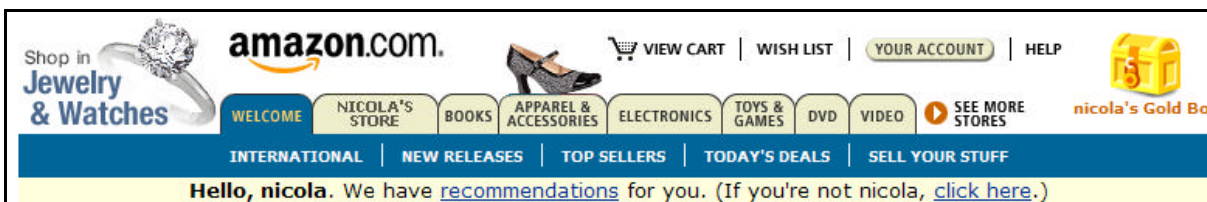


Gestione del carrello

L'applicazione oltre all'inserimento, però, deve fornire anche la gestione del carrello, deve cioè cercare di imitare una comune pratica di acquisto mediante:

- Cancellazione di un prodotto.
- Modifica quantità.
- Eliminazione completa.
- Visualizzazione del contenuto.
- Memorizzazione del carrello (opzionale).

Alla gestione si accede mediante un opportuno collegamento, spesso in forma iconica come nella figura⁴⁹ che segue:



Implementazione del carrello

L'implementazione del carrello può essere effettuata in diversi modi e la scelta dipende dagli scopi dell'applicazione oltre che dalle preferenze del programmatore per l'utilizzo di alcuni oggetti/tecniche del linguaggio rispetto ad altri.

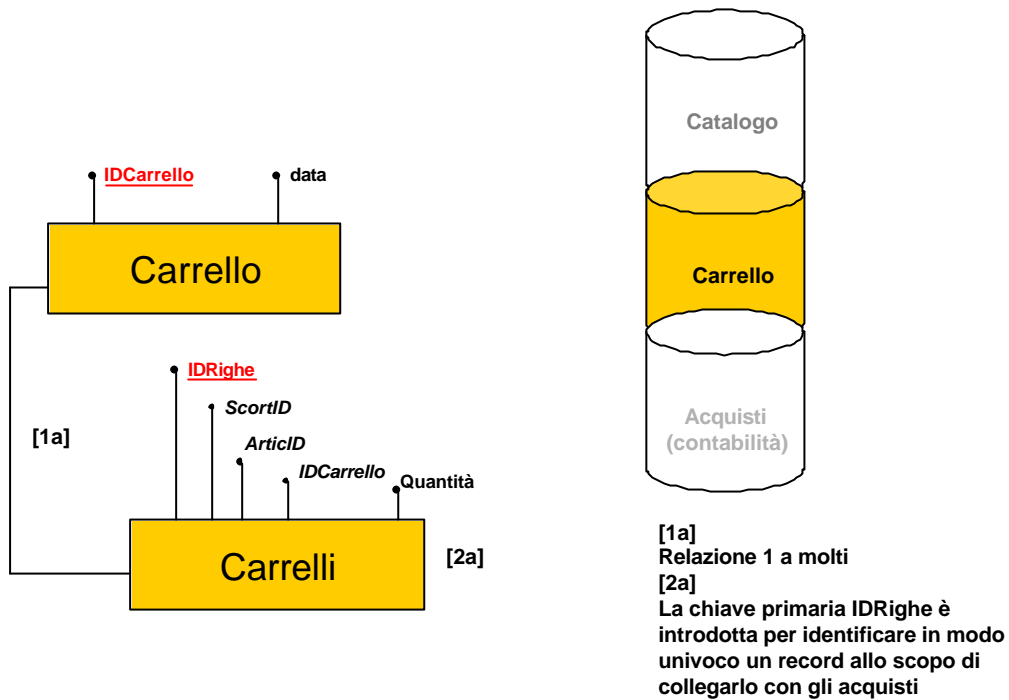
L'implementazione prevede:

- Assegnazione automatica del carrello ad un utente quando entra nel sito, indipendentemente dalla pagina di accesso: sul lato server mediante registrazione nel database e sul lato client grazie alla conservazione del codice del carrello, reso disponibile in modo temporaneo⁵⁰ a tutte le pagine utilizzando un Cookie temporaneo o eventualmente un parametro di Sessione.
- Memorizzazione del carrello utilizzando due tabelle in relazione 1 a molti:
 1. Come unico record in una tabella Carrello contenente semplicemente il codice del carrello e la data di creazione⁵¹.
 2. Memorizzazione del carrello nel database utilizzando un record nella tabella Carrelli, contenente i dati per l'identificazione dei prodotti inseriti nei carrelli da parte di tutti gli utenti (quindi ogni record rappresenta un prodotto assegnato ad un preciso carrello).

⁴⁹ Immagine da www.amazon.com

⁵⁰ Un'altra soluzione più complessa prevede di rendere disponibile in modo permanente il codice del carrello per permettere all'utente di conservare i prodotti inseriti nel carrello per futuri acquisti. In questo caso il carrello deve essere recuperato e riassegnato all'utente che torna il che comporta un sistema di identificazione con un sistema di richiesta esplicita del suo carrello.

⁵¹ Campo che serve per eliminare dalla tabella i carrelli vecchi.



Le tabelle possono per esempio contenere i seguenti dati, compatibili con la struttura:

Carrello

IDCarrello	Data
1	4/10/2004
2	4/10/2004
3	4/10/2004
4	5/10/2004
5	5/10/2004

Carrelli

IDRighe	ScortID	ArticolID	IDCarrello	Quantita
1	22	3	4	1
2	21	3	5	1
3	2	5	5	1
4	38	4	5	1

Il carrello, identificato dal codice pari a 5, che viene visualizzato contiene i dati rilevanti per il cliente e quindi questi è ricostruito come risultato di una query alla tabella Carrelli per la chiave codice del carrello e alle altre tabelle del catalogo.

L'utente visualizza:

Descrizione	Taglia	Colore	Prezzo	Quantita
Scarpe da barca	40	blu	30,00 €	1
Sacco a pelo	Sagomato	Azzurro	360,00 €	1
Parka	L	Rosso	210,00 €	1

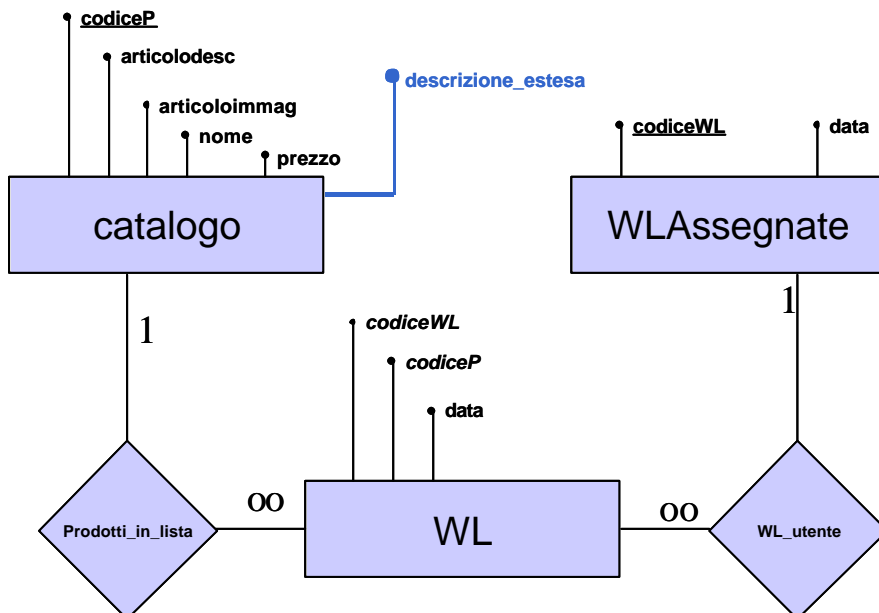
L'implementazione si complica se si aumentano le possibilità per l'utente, cioè i servizi di shopping.

Servizi optional

Un esempio di servizi di shopping è la possibilità di avere più carrelli oppure la possibilità di definire un carrello base, cioè un carrello che ha già inserito dei prodotti che l'utente acquista reiteratamente.

Un altro servizio di shopping utilizzato è la disponibilità di una Wish List (WL) il cui scopo è di consentire ad un utente, in modo facoltativo e senza operazioni di registrazione, di memorizzare un prodotto del catalogo che ritiene interessante in una propria personale lista dei desideri per eventuali acquisti futuri.

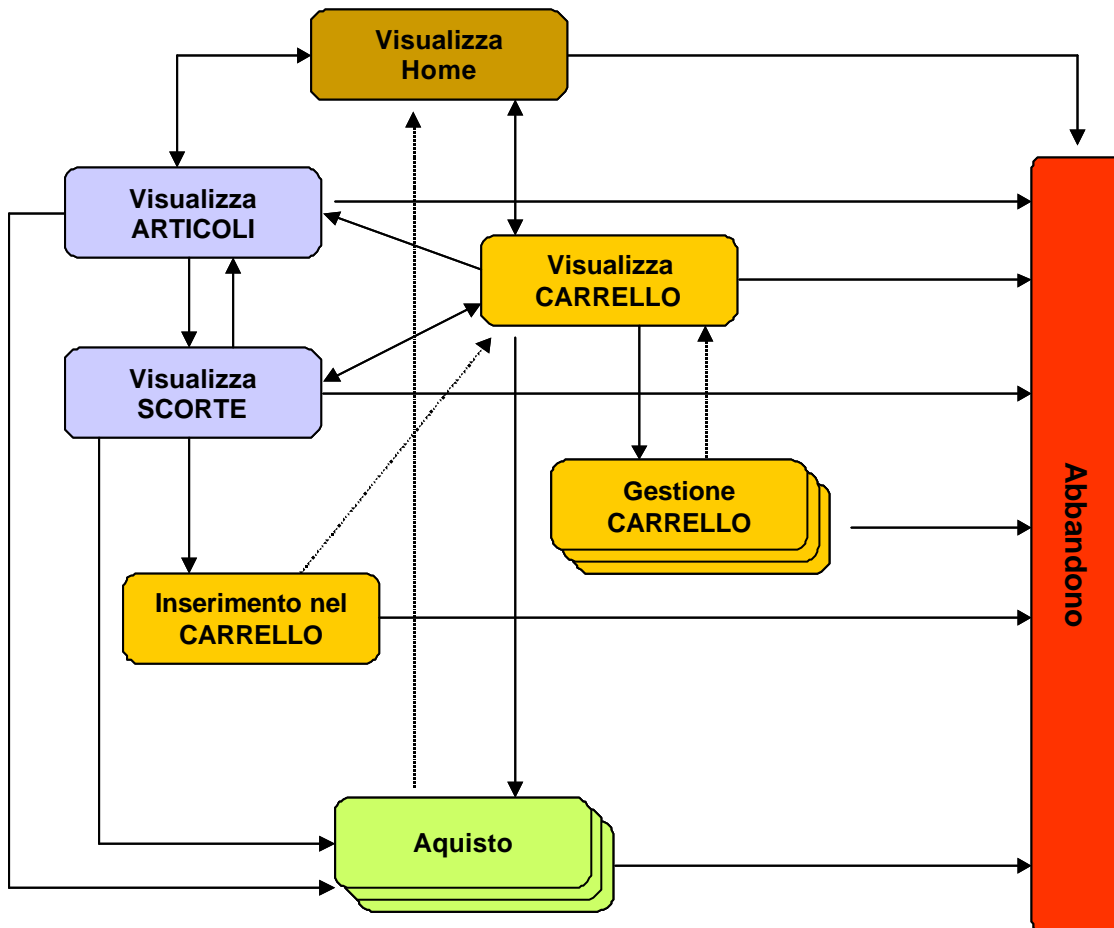
Nel caso della Wish List l'applicazione deve permettere all'utente di trasferire dei prodotti (o anche tutti) da questa al carrello per l'eventuale acquisto⁵².



⁵² Nel caso della WL il processo d'acquisto prevede un ulteriore passo: WL > Carrello > Acquisto.

Schema dei task

Dopo aver analizzato i singoli sottoprocessi possiamo rappresentare con un diagramma lo schema dei task che un utente può fare con l'applicazione/sito.



L'acquisto

Il processo di acquisto è il momento più delicato e consiste tipicamente nei seguenti task:

1. **Autenticazione per l'acquisto (Sign In):** vale quanto detto in precedenza, cioè la necessità di distinguere tra cliente al primo acquisto e cliente già noto. Nel caso di primo acquisto le informazioni richieste devono essere quelle strettamente necessarie, si deve cioè evitare di richiedere un numero enorme di dati come hobby, preferenze...
2. **La scelta della modalità di consegna** in termini di tempo e spedizionario (Shipping).



3. **Il pagamento** con la scelta della modalità e nel caso di pagamento online con carta di credito o elettronica l'inserimento dei dati della carta (Payment).
4. **Confezionamento** con possibilità di personalizzare il regalo (Gift).
5. **Indirizzo di spedizione**, può essere diverso dall'indirizzo dell'acquirente come nel caso di un regalo.



6. **Invio dell'ordine**⁵⁵, mediante Form.

⁵³ Immagine da www.amazon.com

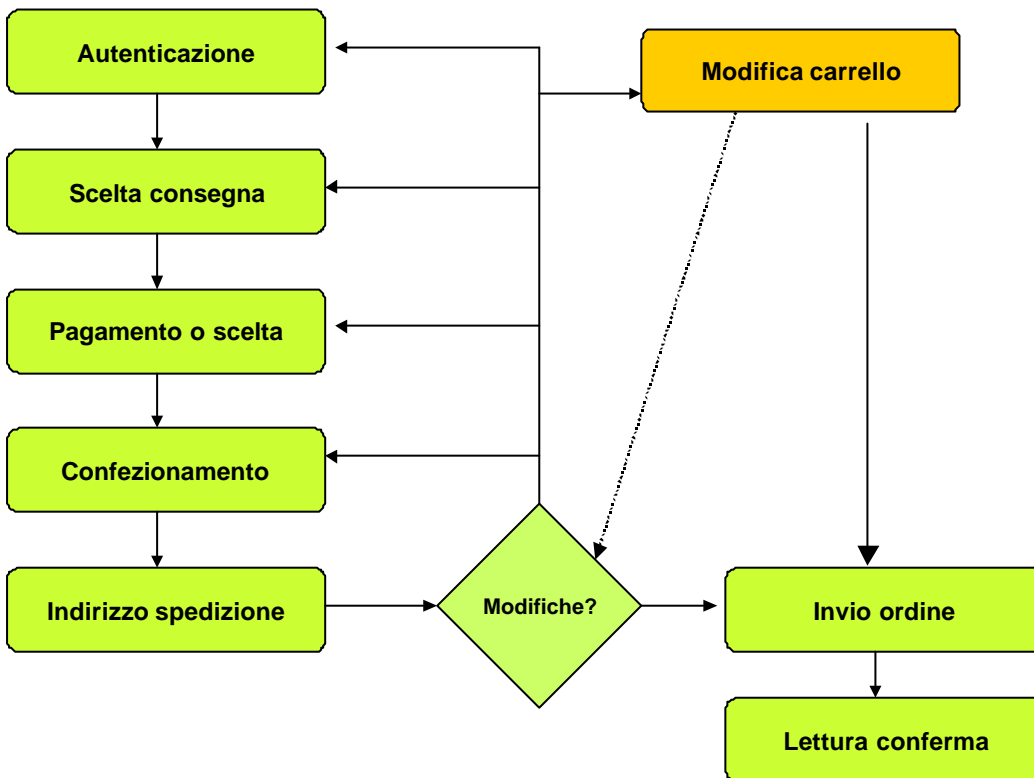
⁵⁴ Immagine da www.amazon.com

⁵⁵ In alcuni casi dopo l'invio si chiede una ulteriore conferma.

7. **Letture della conferma dell'ordine attraverso Email**, quindi invio automatico dell'applicazione web dell'email utilizzando un server di Email.

È importante sottolineare che in ogni fase dell'acquisto si deve fornire la possibilità di modificare le scelte fatte in precedenza compreso anche l'ordine.

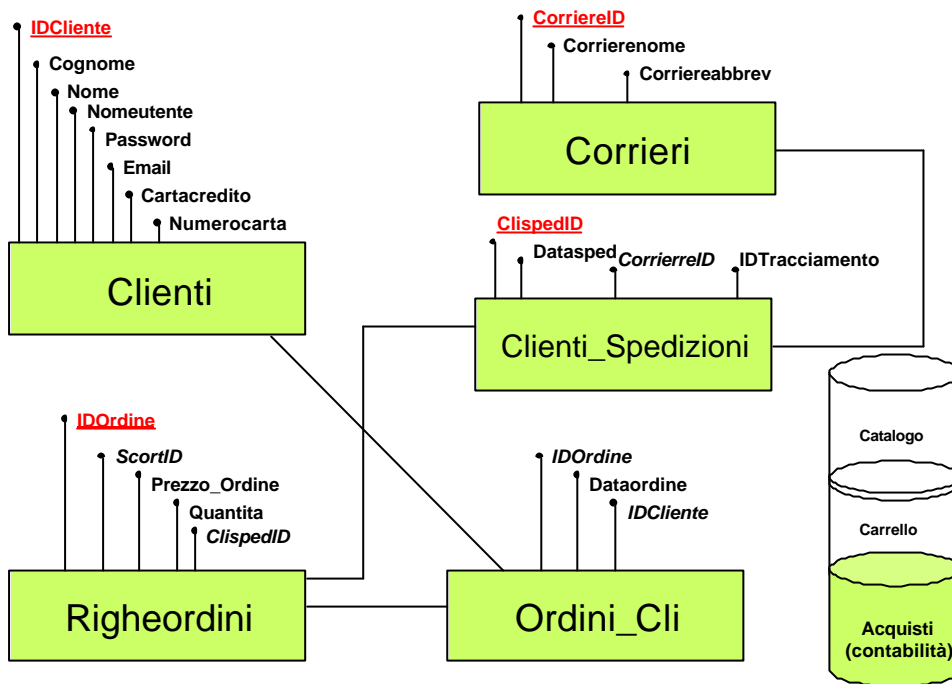
Tenendo conto anche della possibilità della modifica il processo può essere rappresentato in questo modo:



È con il processo di acquisto che la parte del database classico di una azienda, cioè quello della gestione dei clienti e degli ordini si lega con il database del carrello e anche del catalogo.

In particolare se l'ordine viene inviato allora i dati inseriti sono consolidati e contabilizzati, sino a quel momento sono temporanei e cancellati nel caso di ordine non inviato, ad eccezione dei dati del cliente che rimangono per futuri acquisti permettendo l'acquisto rapido con 1-Click.

Il database risulta essere:

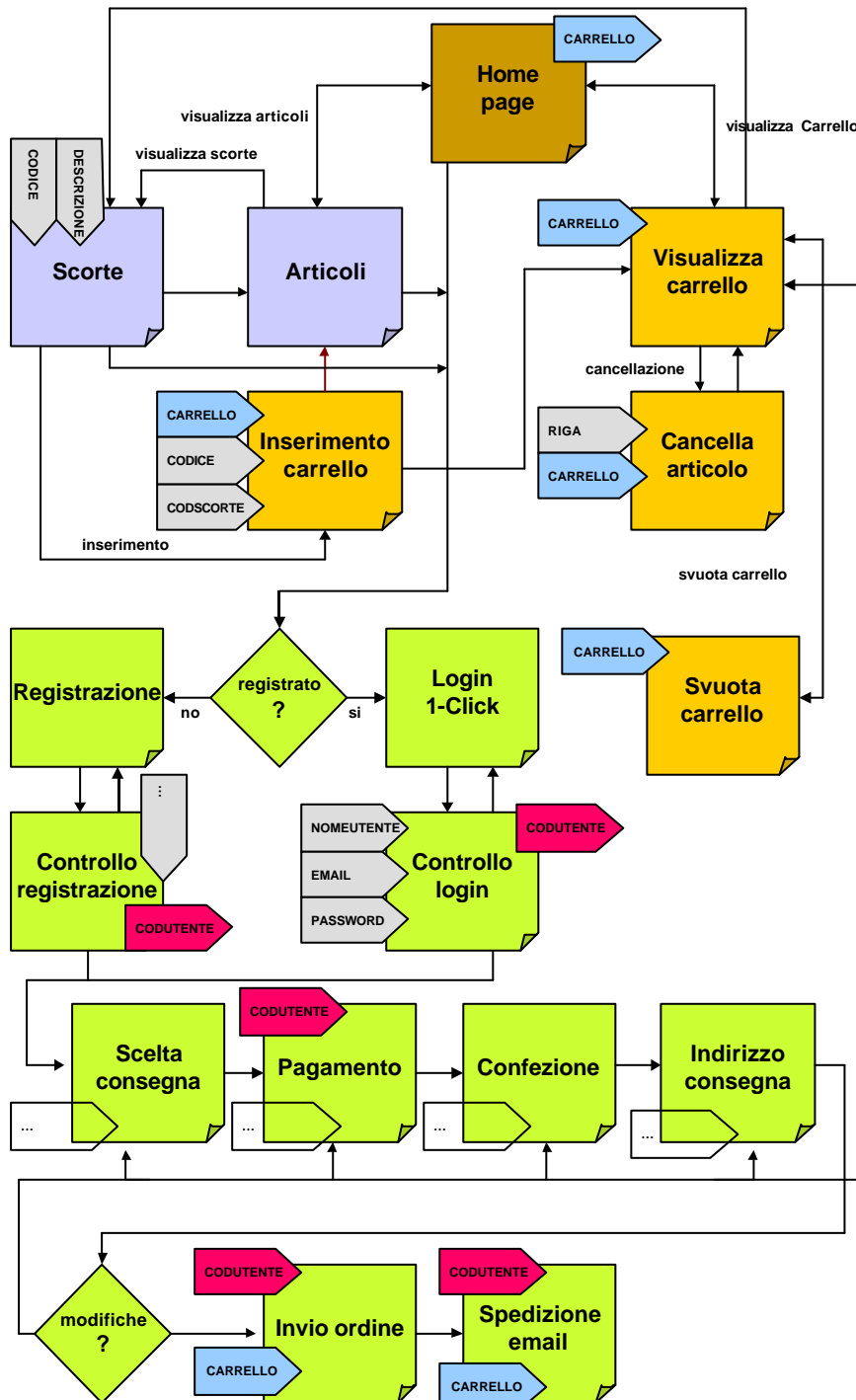


Da notare:

- L'utilizzo della tabella Righeordini che raccoglie tutti i prodotti ordinati di tutti i clienti.
- La tabella Ordini_Cli che permette di collegare i prodotti al cliente generando l'ordine del cliente.
- Le tabelle Clienti-Spedizioni e Corrieri che permettono di associare il prodotto da spedire allo spedizioniiere e di memorizzare le informazioni sulle spedizioni, tra cui il codice di tracciamento IDTracciamento (che permette collegandosi al sito dello spedizioniiere di vedere lo stato della spedizione).
- Ogni singolo prodotto può essere spedito da uno spedizioniiere diverso.
- Non c'è un collegamento tra gli ordini e il carrello dell'utente e dalla tabella dei carrelli si prelevano il codice delle scorte, ScortID, e la quantità.

Relazioni tra le pagine

Lo schema delle relazioni tra le pagine con la parte dell'acquisto incompleta per quanto riguarda i parametri potrebbe essere:



Script pagine

Si presentano gli script di alcune pagine:

- Home page.
- Visualizza articoli.
- Visualizza scorte.
- Inserimento carrello.
- Visualizza carrello.

Per meglio comprendere gli script si riporta il contenuto di alcune pagine:

1. Visualizza articoli del catalogo.

N	Cod	Descrizione	Categoria	Sesso	Promozione
1	1	Pantaloncini sportivi donna	Pantalone	Donna	N
2	2	Giaccone sportivo donna	Giaccone	Donna	N
3	3	Scarpe da barca	Scarpe	Uomo	N
4	4	Parka	Giaccone	Uomo	S
5	5	Sacco a pelo	Tempo libero	Unisex	S

2. Visualizza scorte (dell'articolo "Giaccone sportivo donna") con la possibilità di inserire nel carrello.

N	Taglia	Colore	Prezzo	Disponibilità	Carrello
1	L	Blu	29,95	115	Inserisci
2	L	Cachi	29,95	<i>non disponibile (informati)</i>	No
3	M	Blu	29,95	137	Inserisci
4	M	Cachi	29,95	147	Inserisci
5	S	Blu	29,95	139	Inserisci
6	S	Cachi	29,95	150	Inserisci

Carrello










- [Visualizza](#)
Visualizza il contenuto del carrello.

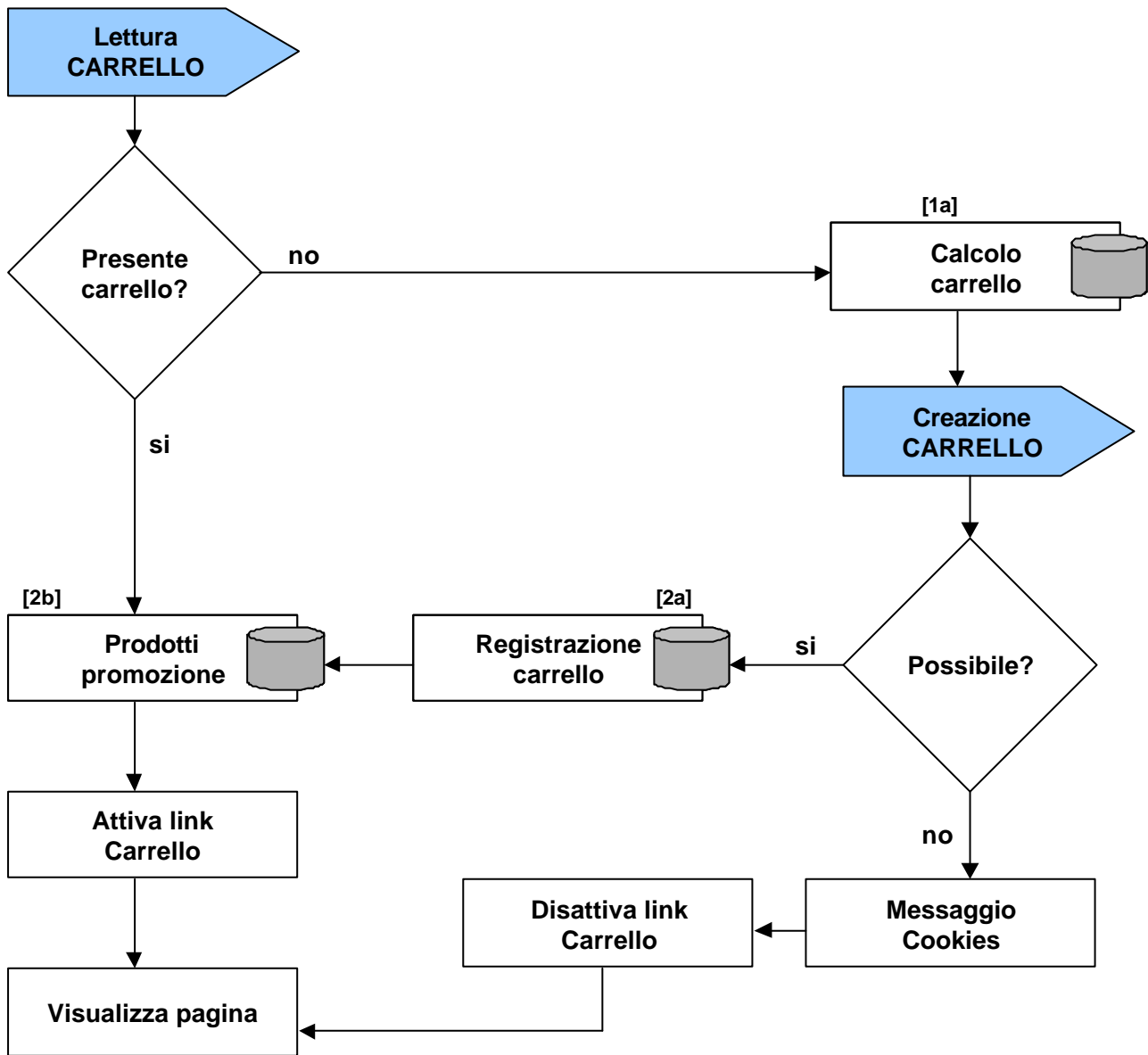
3. Visualizza carrello semplificato.

N	Descrizione	Taglia	Colore	Q	Operazione
1	Parka	M	Nero	1	Elimina
2	Pantaloncini sportivi donna	L	Blu	1	Elimina

Carrello

- [Svuota](#)
Elimina completamente e definitivamente i prodotti inseriti nel tuo carrello

Home page



[1a]

Estrazione dalla tabella Carrello dell'ultimo codice inserito e calcolo del nuovo codice con incremento di 1.

```
SELECT MAX() AS UltimoCarrelloInserito FROM Carrello
```

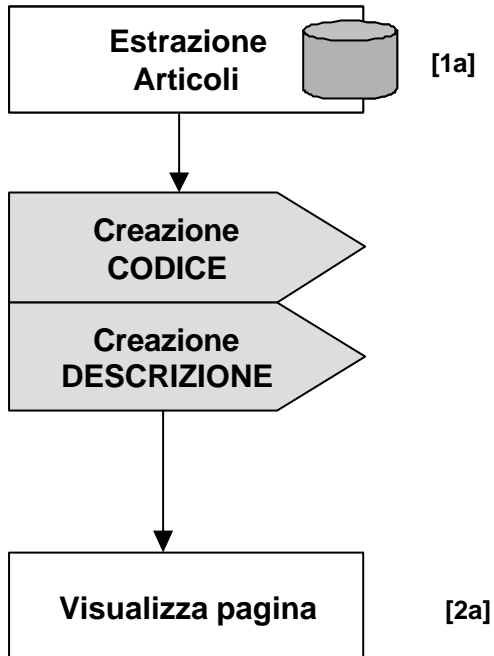
[2a]

```
INSERT INTO Carrello () VALUES ()
```

[2b]

```
SELECT * FROM ARTICOLI WHERE Promozione='S'
```

Visualizza articoli



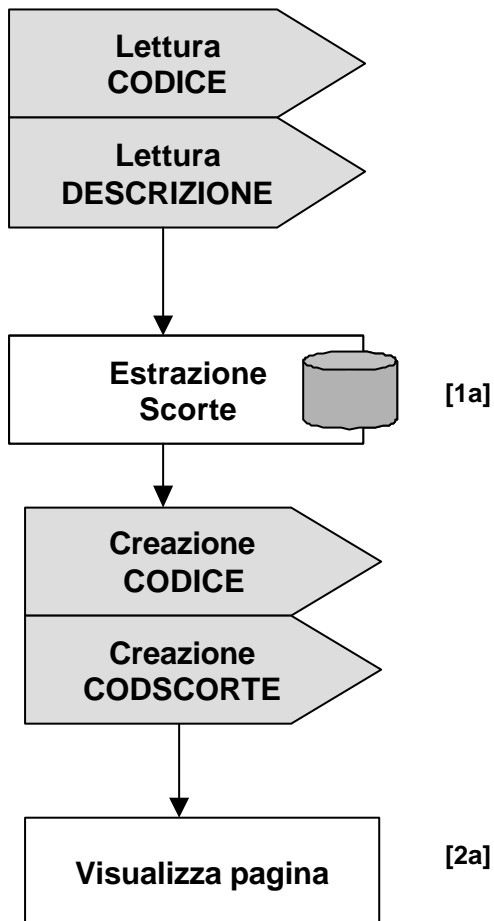
[1a]

Estrazione dalla tabella Articoli dei prodotti
`SELECT * FROM Articoli`

[2a]

Nella visualizzazione della pagina si genera il collegamento URL accodando alla pagina che visualizza le scorte i parametri CODICE che corrisponde al valore del campo ArticolID e DESCRIZIONE che corrisponde al valore del campo Articolodesc.

Visualizza scorte



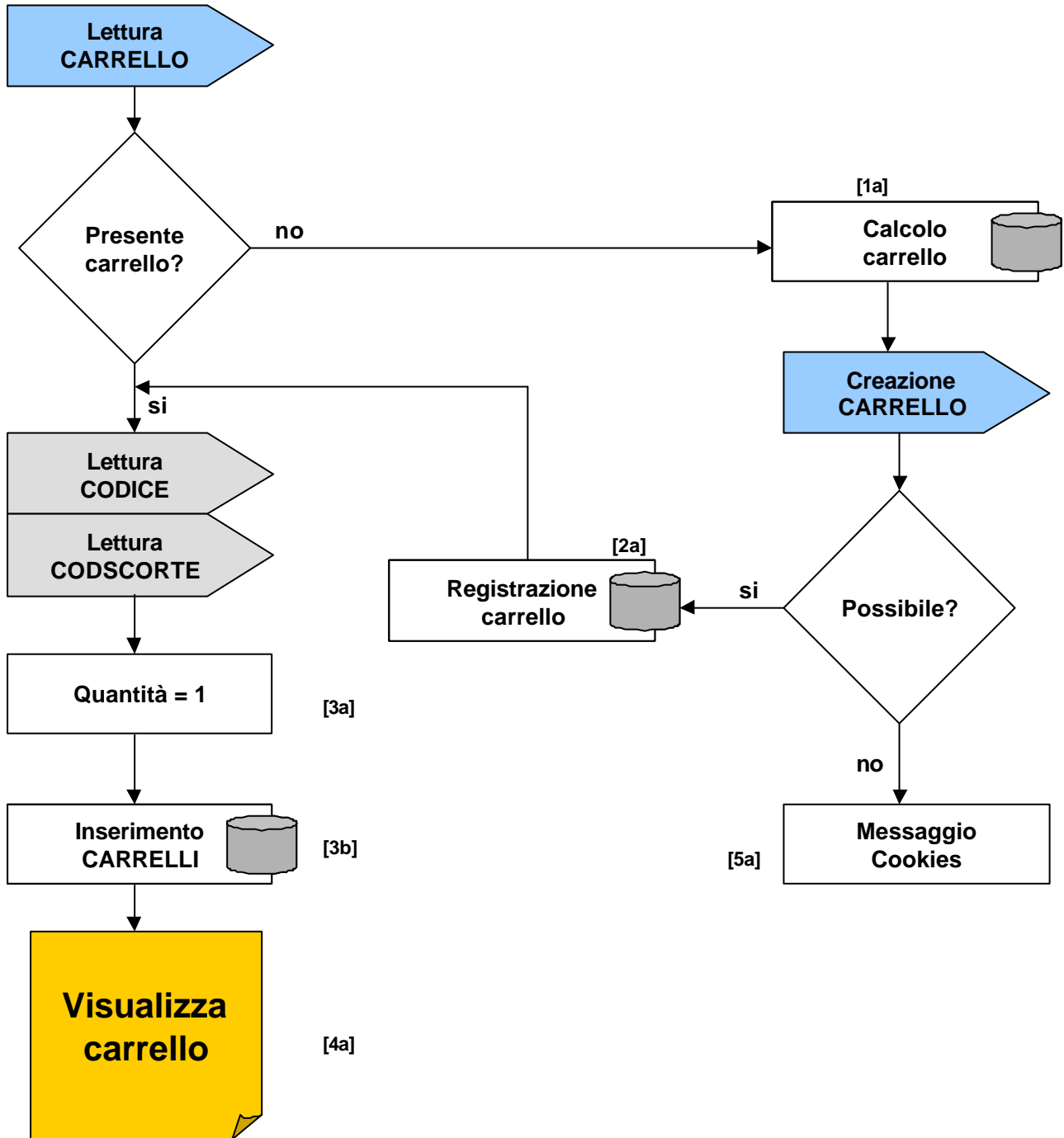
[1a]

Estrazione dalla tabella Scorte delle configurazioni dell'articolo
`SELECT * FROM Scorte WHERE ArticID = CODICE`

[2a]

Nella visualizzazione della pagina si genera il collegamento URL accodando alla pagina che inserisce il prodotto nel carrello i parametri CODICE che corrisponde al valore del campo ArticID e CODSCORTE che corrisponde al valore del campo ScortID.

Inserimento nel carrello



[1a]
Estrazione dalla tabella Carrello dell'ultimo codice inserito e calcolo del nuovo codice con incremento di 1.
SELECT MAX() AS UltimoCarrelloInserito FROM Carrello

[2a]

INSERT INTO Carrello () VALUES ()

[3a]

L'utente non può stabilire la quantità da acquistare che quindi viene automaticamente impostata a 1 quando si attiva l'inserimento nel carrello.

[3c]

Estrazione dalla tabella Carrelli dell'ultimo codice inserito e calcolo del nuovo codice con incremento di 1.

SELECT MAX() AS UltimaRigaInserita FROM Carrelli

[3b]

INSERT INTO Carrelli () VALUES ()

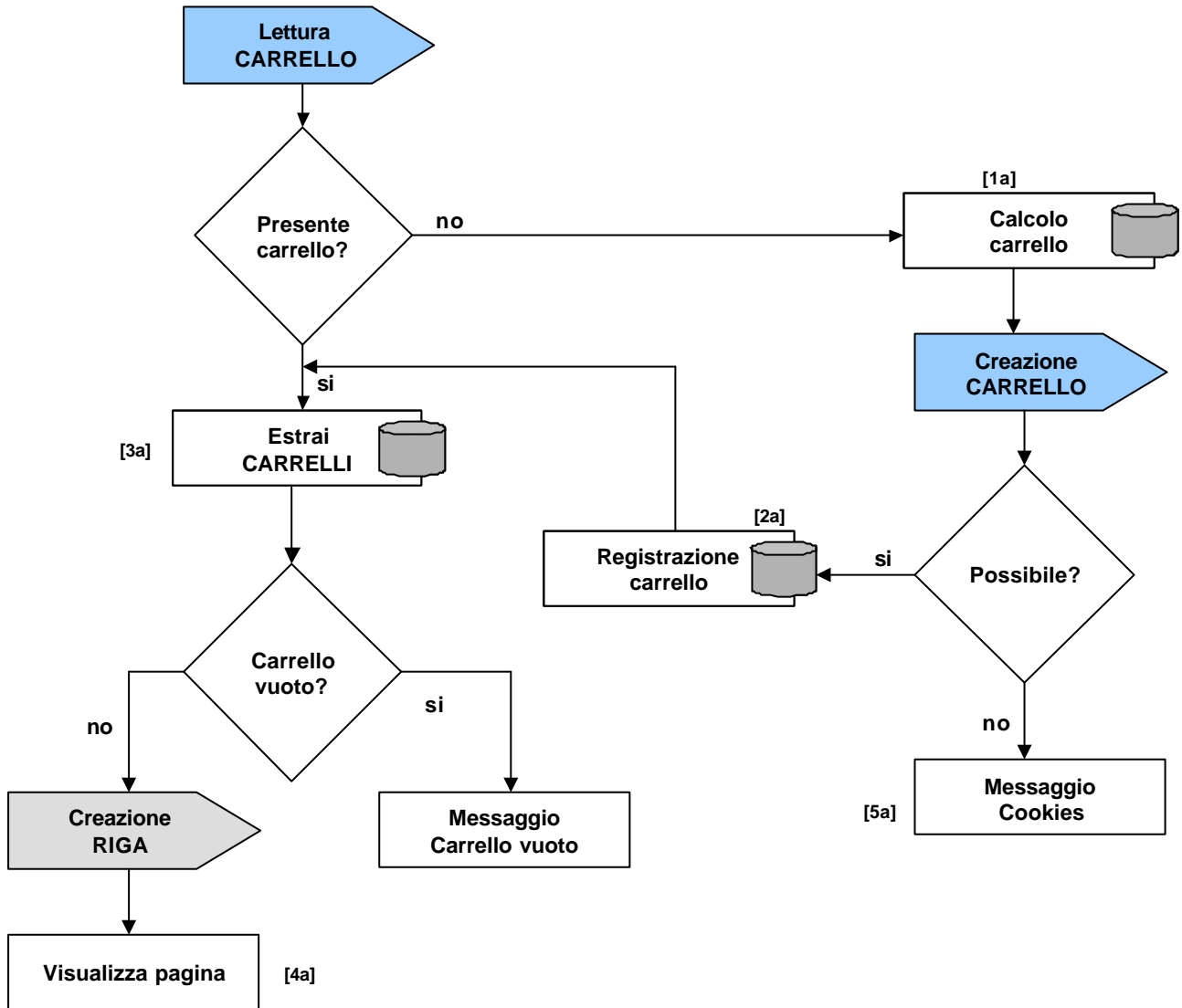
[4a]

Reindirizzamento alla pagina.

[5a]

Nella pagina si visualizza il messaggio che l'utilizzo del sito comporta l'abilitazione alla ricezione dei cookies.

Visualizza carrello



[1a]

Il carrello è costruito partendo dal codice del carrello assegnato con il cookie CARRELLO e mediante una interrogazione alla tabella Carrelli, Articoli e Scorte.

```
SELECT MAX() AS UltimoCarrelloInserito FROM Carrello
```

[2a]

```
INSERT INTO Carrello () VALUES ()
```

[3a]

Il carrello è costruito partendo dal codice del carrello assegnato con il cookie CARRELLO e mediante una interrogazione alla tabella Carrelli, Articoli e Scorte.

```
SELECT idrighe, articolodesc, artictaglia, colore, prezzo_corr, quantita FROM carrelli, articoli, scorte WHERE carrelli.idcarrello = CARRELLO AND carrelli.scortid = scorte.scortid AND articoli.articid = scorte.articid ORDER BY idrighe DESC
```

[4a]

Nella visualizzazione della pagina si genera il collegamento URL accodando alla pagina che inserisce elimina il prodotto dal carrello il parametro RIGA che corrisponde al valore del campo RigaID della tabella Carrelli.

[5a]

Nella pagina si visualizza il messaggio che l'utilizzo del sito comporta l'abilitazione alla ricezione dei cookies.

Codice ASP

Riportiamo parte del codice in ASP di due pagine:

1. Inserimento nel carrello.
2. Visualizza carrello.

Inserimento nel carrello

```
<%
Option Explicit
Response.Buffer = True
Response.Expires = 0
Dim strQuery, intIddoc, rsProdotti , intConta, intCodice, strDescrizione, intCarrello,
rsUltimoCarrello,intScorte
Set connDB = Server.CreateObject("ADODB.Connection")
%>
<!-- #INCLUDE FILE="conndatabase.inc"--> 56
<%
connDB.open(strPercorso)
If Request.Cookies("CARRELLO") = "" Then

    Dim strInserimento
    strQuery = "SELECT MAX(IDCARRELLO) AS ultimo FROM carrello"
    set rsUltimoCarrello = connDB.Execute(strQuery)
    Session("CARRELLO") = rsUltimoCarrello("ULTIMO") + 1
    intCarrello = rsUltimoCarrello("ULTIMO") + 1
    strInserimento = "INSERT INTO carrello (idcarrello, data) VALUES (" & intCarrello & "," &
    date() & ")"
    connDB.Execute(strInserimento)

End If
intCarrello = Request.Cookies("CARRELLO")
intCodice = Request.QueryString("CODICE")
intScorte = Request.QueryString("CODSCORTE")
Dim intRiga,rsUltimaRiga
strQuery = "SELECT MAX(IDRIGHE) AS righe FROM carrelli"
set rsUltimaRiga = connDB.Execute(strQuery)
intRiga = rsUltimaRiga("RIGHE") + 1
strInserimento = "INSERT INTO carrelli (idrighe,artid,idcarrello,scortid,quantita) VALUES (" & intRiga
& "," & intCodice & "," & intCarrello & "," & intScorte & ",1)"
connDB.Execute(strInserimento)
ConnDB.Close
Response.Redirect("visualizza.asp")
%>
```

⁵⁶ Il file di connessione al database conndatabase.inc contiene:

```
<%
Dim connDB, strDriver, strPercorso
Set connDB = Server.CreateObject("ADODB.Connection")
strDriver = "Driver={Microsoft Access Driver (*.mdb)};"
strPercorso = " DBQ=" & server.MapPath("prodotti.mdb")
strPercorso = strDriver & strPercorso
%>
```

Visualizza carrello

```

<%
Option Explicit
Response.Buffer = True
Response.Expires = 0
Dim strQuery, intIddoc, rsProdotti , intConta, intCodice, strDescrizione, intCarrello,
rsUltimoCarrello,intScorte
%>
<!-- #INCLUDE FILE="conndatabase.inc"-->
<% connDB.open(strPercorso) %>
<html>
...

<%
If Request.Cookies("CARRELLO") = "" Then
%>

    <p class="testo">
    Non hai nessun carrello assegnato che ti verrà automaticamente assegnato quando effettuerai
    l'isperimento di un prodotto dal <a href="catalogo.asp">catalogo</a> .
    </p>

<% Else %>

    <%
    intCarrello = Request.Cookies("CARRELLO")
    strQuery="SELECT idrighe, articolodesc, artictaglia, colore, prezzo_corr, quantita FROM
    carrelli, articoli, scorte WHERE carrelli.idcarrello = " & intCarrello & "AND carrelli.scortid =
    scorte.scortid AND articoli.articid = scorte.articid ORDER BY idrighe DESC"
    Set rsProdotti = connDB.Execute(strQuery)
    %>

    <p class="par">Contenuto del carrello (codice <%=intCarrello %>)</p>

    <% If rsProdotti.EOF Then %>

    <p class="testo">Il carrello non contiene prodotti.</p>

    <% Else %>

        <div align="center">
        <table border="0" cellpadding="5" cellspacing="1" width="85%" bgcolor="black">
        <tr>
        <td bgcolor="#eee8aa" align="center" width="4%">
        <p class="campoform"><b>N</b></p>
        </td>
        <td bgcolor="#eee8aa" width="40%" align="center">
        <p class="campoform"><b>Descrizione</b></p>
        </td>
        <td bgcolor="#eee8aa" align="center">
        <p class="campoform"><b>Taglia</b></p>
        </td>
        <td bgcolor="#eee8aa" align="center">
        <p class="campoform"><b>Colore</b></p>
        </td>

```

```

<td bgcolor="#eee8aa" align="center">
<p class="campoform"><b>Q</b></p>
</td>
<td bgcolor="#ffc0cb" align="center">
<p class="campoform"><b>Operazione</b></p>
</td>
</tr>
<%
intConta = 0
Do While NOT rsProdotti.EOF
intConta = intConta + 1
%>
<tr>
<td align="center" width="4%" bgcolor="white"><p class="campoform"><% =intConta
%>
</p></td>
<td width="40%" align="center" bgcolor="white">
<p class="campoform"><% =rsProdotti("ARTICOLODESC") %></p></td>
<td align="center" bgcolor="white">
<p class="campoform"><% =rsProdotti("ARTICTAGLIA") %></p></td>
<td align="center" bgcolor="white"><p class="campoform"><% =rsProdotti("COLORE")
%> </p></td>
<td align="center" bgcolor="white">
<p class="campoform"><% =rsProdotti("QUANTITA") %></p><td>
<td align="center" bgcolor="#ffc0cb">
<p class="campoform">
<a href="elimina.asp?RIGA=<% =rsProdotti("IDRIGHE") %>">
<b>Elimina</b></a></p>
</td>
</tr>
<% rsProdotti.MoveNext %>
<% Loop %>
<tr>
<td bgcolor="#ffc0cb" align="left" colspan="6" valign="top"
background="imm/carrello.gif">
<p class="campoform"><b>Carrello</b></p>
<ul>
<li type="square" class="campoform">
<a href="svuota.asp?CAR=<% =intCarrello %>"><b>Svuota</b></a><br>
Elimina completamente e definitivamente i prodotti inseriti nel tuo carrello
</td>
</tr>
<% connDB.Close %>
</table>
<p></p>
</div>

<% End If %>

```

```

<% End If %>

```

```

...
</html>

```

Bibliografia

Michele Visciola, Usabilità dei siti Web, Apogeo 2001, Milano

Carlo Pelliccia, ASP la guida introduttiva I, Edizioni Master 2002, Milano

Carlo Pelliccia, ASP la guida introduttiva II, Edizioni Master 2002, Milano

Jeremy Rifkin, L'era dell'accesso, Saggi Mondadori 2000, Milano

Sofia Postai, Siti che funzionano, Apogeo Milano, 2001

Autori vari, Guida al commercio elettronico II edizione, INDIS e Ministero dell'Industria e del Commercio 1999, Roma

Patrick j. Lynch e Sarah Hoerton, Web Style Guide 2nd edition , Yale University Press 2002, New Have

Sitografia

www.html.it

www.aspitalia.it

www.microsoft.com/italy/scripting/default.html

www.webstyleguide.com

www.informationarchitecture.it

Note

Questo documento che nasce dalla sistematizzazione di alcune dispense per gli studenti dell'indirizzo tecnologico informatica e per i corsi del fondo sociale europeo.

Lo scopo, comunque, non è quello di esaurire l'argomento (ammesso che lo sia), ma fornire una cornice di riferimento per futuri approfondimenti.

Si prega, infine, di non diffondere ad altri il documento per ovvi motivi legali (non miei, ma degli autori a cui ho preso a prestito e non ancora giustamente citati...).

Nicola Ceccon

Indice

Basic concepts	pagina 2
Tutorial: applicazione di ecommerce	pagina 27
Tutorial: il progetto dell'applicazione di ecommerce	pagina 33
Codice ASP	pagina 56
Bibliografia	pagina 60
Sitografia	pagina 60
Note	pagina 60

Titolo:	Applicazione web
Autore:	Nicola Ceccon
Email:	nicola.ceccon@istruzione.it
Classe:	QUINTA INFORMATICA (5IA)
Anno scolastico:	2004/2005
Scuola:	Itis Euganeo Via Borgofuro 6 - 35042 Este (PD) - Italy Telefono 0429.21.16 - 0429.34.72 Fax 0429.41.86 http://www.itiseuganeo.it informazioni@itiseuganeo.it
Note legali:	Diffusione consentita con obbligo di citarne le fonti