



Eleworld2018

System Design Document

Daniele Cioffi

Mario Consalvo

Indice

1. Introduzione

- 1.1 Scopo del sistema**
- 1.2 Obbiettivi di progettazione**
- 1.3 Descrizioni, Acronimi, Abbreviazioni**
- 1.4 Riferimenti**
- 1.5 Panoramica**

2. Architettura software corrente

3. Architettura software proposta

- 3.1 Panoramica**
- 3.2 Decomposizione del sottosistema**
- 3.3 Software/hardware mapping**
- 3.4 Gestione dei dati persistenti**
- 3.5 Access control and security**
- 3.6 Global software control**
- 3.7 Boundary conditions**

4. Architettura sottosistema

- 4.1 Architettura model**
- 4.2 Architettura view**
- 4.3 Architettura controller**

Coordinatori del progetto

Prof. Andrea DE LUCIA
Prof.ssa Rita FRANCESCE

Partecipanti

Daniele Cioffi – Matricola 0512103446
Mario Consalvo – Matricola 0512103500

Cronologia delle Revisioni

Data	Versione	Descrizione	Autore
08/12/2017	1.0	Prima Stesura	Cioffi Daniele Mario Consalvo
15/12/2017	1.1	Introduzione immagini	Cioffi Daniele Mario Consalvo
20/12/2017	2.0	Stesura Finale	Cioffi Daniele Mario Consalvo

1. Introduzione

1.1 Scopo del sistema

Il progetto Eleworld2018 consiste nella creazione di un sito web e-commerce (Eleworld2018.it) completo di ogni aspetto e funzionalità. Il sito in questione, darà l'opportunità a gli utenti, di visionare e di acquistare, qualsiasi forma di elettrodomestico, in modo pratico e veloce. La finalità del progetto è proprio quella di riuscire a creare un sito web capace di raggiungere traguardi commerciali e di ottenere visibilità sulla rete, ricca di concorrenze.

1.2 Obbiettivi di progettazione

Il principale obiettivo del design di un e-commerce, è rendere semplice e intuitiva la navigazione, per favorire il maggior numero di vendite, mantenendo lo stile e l'immagine aziendale. Molte aziende si trovano impreparate sul fronte dei contenuti perché fino a quel momento questi erano a carico dei rivenditori, oppure non esistevano proprio e quindi sono da creare da zero. I contenuti vanno poi aggiornati costantemente per tenere vivo il sito e per favorirne l'indicizzazione naturale sui motori di ricerca, che premiano i siti attivi e dinamici.

1.3 Descrizioni , acronimi e abbreviazioni

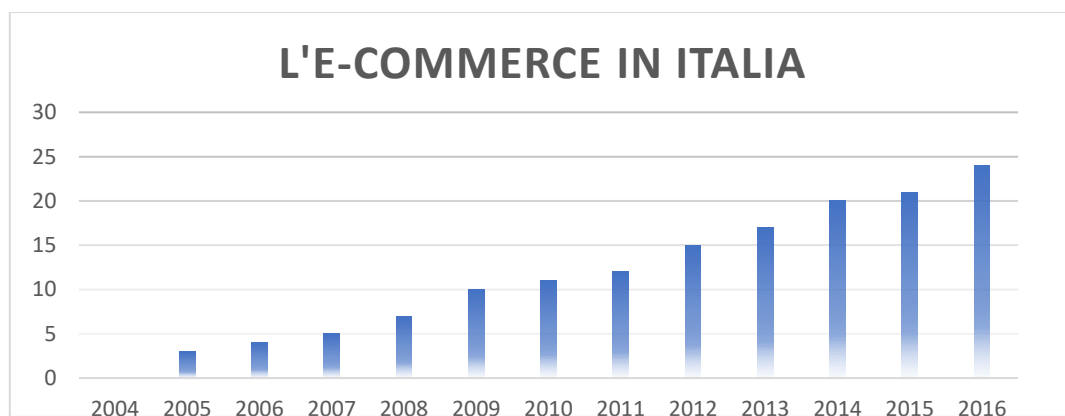
MVC: Model View Control

1.4 Riferimenti

- *Wikipedia.* (<https://it.wikipedia.org>)
- *Florida State University - department of computer science* (<http://www.cs.fsu.edu>)
- *Object-Oriented Software - _Engineering Using UML, Patterns, and Java™ (Third Edition - Bernd Bruegge & Allen H. Dutoit)*

1.5 Panoramica

L'e-commerce è una realtà il cui sviluppo pare non avere limite alcuno. Il valore degli acquisti online degli Italiani ha raggiunto nel 2015 oltre 16,6 miliardi di euro, con un incremento del 16% rispetto al 2014 (pari a 2,2 miliardi di euro). L'intero comparto del commercio elettronico, è trainato da quattro mercati principali: Turismo, Elettronica-Informatica, Abbigliamento, Editoria. Al di là dei pur indubbi risultati acquisiti in merito ai settori maggiori, risulta chiaro come il mercato del commercio elettronico si stia uniformando sempre più ai modelli occidentali più robusti (Stati Uniti, Gran Bretagna e Germania in primis), e che l'acquisto online di prodotti stia progredendo a un ritmo di crescita più elevato rispetto all'acquisto esclusivo di servizi. Prova di questo incremento positivo è data dal fatto che il mercato dei prodotti acquistabili online può contare su un ventaglio di scelte sempre maggiori, coadiuvate dallo sviluppo di varie comparti. Il dinamismo che si accompagna alla definizione di nuovi settori, è legato al fatto che, fino a qualche tempo fa, molte di quelle categorie erano quasi del tutto assenti sul web, e che proprio lo sviluppo della multicanalità, così come il legame sempre più saldo tra online e offline all'interno del percorso d'acquisto degli utenti, ha potuto contribuire a definire, plasmare, modellare meglio l'intero panorama del commercio elettronico, così da permettere a utenti e imprese ad avere una visione più chiara dell'intero fenomeno della vendita online. Non è un mistero che molti attori impegnati nel business, in mancanza di strumenti d'informazione adeguati, avessero sempre scongiurato tale tipo di commercio, evidenziandone le pecche e mettendone a tacere gli evidentissimi vantaggi. Attualmente, invece, la consapevolezza relativa all'e-commerce va aumentando sempre di più, e richiama l'attenzione e il consenso di sempre più imprese legate ancora alla tradizione. Diverse aziende, infatti, nel 2015, hanno attivato un sito di vendita online, affiancandolo talvolta a un marketplace (Amazon e eBay, tra i tanti) per beneficiare di vantaggi aggiuntivi.



2. Architettura software corrente

Affinché un sistema alla base di un e-commerce funzioni alla perfezione è di fondamentale importanza che esista una consistenza dei dati e delle informazioni sui prodotti e sulle transazioni garantita fra i processi e i sistemi multipli che spesso sono alla base delle compravendite e dei sistemi di gestione dei punti vendita. La pratica di retail multicanale o quello onnipresente che coinvolge il punto di vendita fisico, un sito Web e un accesso mobile per smartphone e tablet deve quindi contare su un sistema informatico e informativo capace di passare le informazioni da un punto all'altro dell'intero processo di gestione, senza incongruenze in modo da agevolare l'esperienza utente e permettere davvero di acquistare su un canale ed entrare in possesso del bene tramite un altro metodo. Queste architetture IT enterprise devono essere capaci di gestire le informazioni necessarie alla vendita online con contenuti aggiuntivi che normalmente non servono nella vendita in negozio. Così se in negozio è sufficiente un codice a barre o uno SKU, un nome prodotto e una breve descrizione, online sono necessarie foto di qualità del bene in vendita, contenuti e descrizioni approfondite, commenti e recensioni pensate per migliorare i tassi di conversione, contributi multimediali di ogni genere e tutti gli attributi necessari per consentire agli utenti di individuare il prodotto nel mare magnum del Web. Il sistema IT per l'e-commerce multicanale deve quindi saper gestire i diversi aspetti delle vendite, creando così un sistema cross-funzionale e di comunicazioni interdipartimentali capace di far intervenire l'addetto alla vendita, al magazzino, al rifornimento, al marketing e altre tipologie di figure ogni volta che se ne renda necessario per aiutare l'utente a concludere l'acquisto sui diversi canali di approvvigionamento del bene.

La vendita vera e propria, si realizza tramite piattaforma web, su cui il venditore carica il catalogo prodotti/servizi così che il compratore lo possa consultare online, scegliendo i prodotti da acquistare e inviando l'ordine. A seconda del profilo cliente si parla di:

- Business to Consumer (B2C): azienda che vende a un privato.
- Business to Business (B2B): azienda che vende a un'altra azienda.

Entrando più nello specifico, le principali piattaforme per creare un sito e-commerce sono numerose e, tutte multilingue e in grado di gestire più valute. Ne elenchiamo alcune:

Joomla VirtueMart

Parliamo di una piattaforma free, compatibile con le ultime versioni di Joomla. Ha tutto quello che serve per realizzare e amministrare un negozio virtuale, come gli strumenti di gestione degli account e degli indirizzi o lo storico ordini. Inoltre, ci offre la possibilità di inserire un numero illimitato di prodotti e categorie.

Magento

Tra i software e-commerce più sfruttati al mondo dal suo arrivo nel 2007, dopo 4 anni è stato acquisito da eBay Inc., di conseguenza è perfettamente integrato con eBay e PayPal. Disponibile in due versioni, a pagamento oppure open-source, è adatto ai piccoli come ai grandi negozi, con migliaia di prodotti e accessi giornalieri. Offre funzionalità avanzate di amministrazione del catalogo prodotti, delle spedizioni, dei pagamenti e della reportistica.

OpenCart

È uno dei prodotti più giovani del mercato, non richiede grandi competenze informatiche, è robusto e facile da configurare. Può adattarsi alle esigenze di vendita di chiunque, anche se è rivolto principalmente alle piccole/medie realtà. Con OpenCart possiamo inserire un numero illimitato di prodotti e scegliere tra diversi tipi di pagamento e spedizione

OsCommerce

Tra le prime piattaforme open source nate per creare un sito e-commerce, è stata lanciata nel 2000 in Germania. Oggi la sua community ospita oltre 270.000 membri. È caratterizzata dall'offerta di circa 7000 add-on che la rendono potente e flessibile. Il suo vantaggio principale è la versatilità, ma non bisogna dimenticare che, rispetto ad altri software e-commerce più recenti, ha avuto qualche problema di sicurezza e che in alcuni casi rende difficile la separazione tra dati, struttura e contenuti.

3. Architettura Software proposta

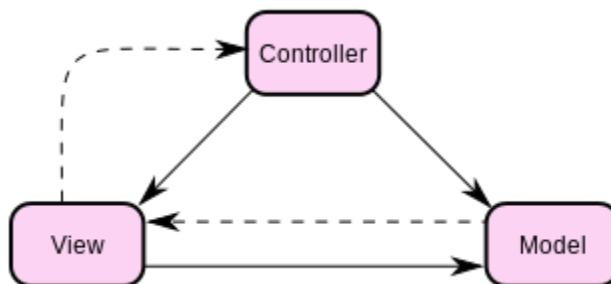
3.1 Panoramica

La realizzazione del sito eleworld2018 con annessa architettura di gestione, verrà realizzata con un'architettura MVC. L'architettura verrà realizzata con la combinazione di servlet e jsp. Il componente centrale del MVC, il modello, cattura il comportamento dell'applicazione in termini di dominio del problema, indipendentemente dall'interfaccia utente. Il modello gestisce direttamente i dati, la logica e le regole dell'applicazione. Una vista può essere una qualsiasi rappresentazione in output di informazioni, come un grafico o un diagramma. La terza parte, il controller, accetta l'input e lo converte in comandi per il modello e/o la vista. Inoltre verrà utilizzato Apache Tomcat, un application server nella forma di servlet container che implementa le specifiche Java Server Pages (JSP) e Servlet, fornendo quindi una piattaforma software per l'esecuzione di applicazioni Web sviluppate in linguaggio Java.

3.2 Decomposizione del sottosistema

I sottosistemi dell'MVC sono così classificati:

- Sottosistema modello: responsabile per il dominio dell'applicazione conoscenza;
- Sottosistema visualizzazione: responsabile della visualizzazione dell'applicazione degli oggetti di dominio per l'utente;
- Sottosistema controllo: responsabile della sequenza di interazioni con l'utente e notifica delle visualizzazioni delle modifiche del modello.



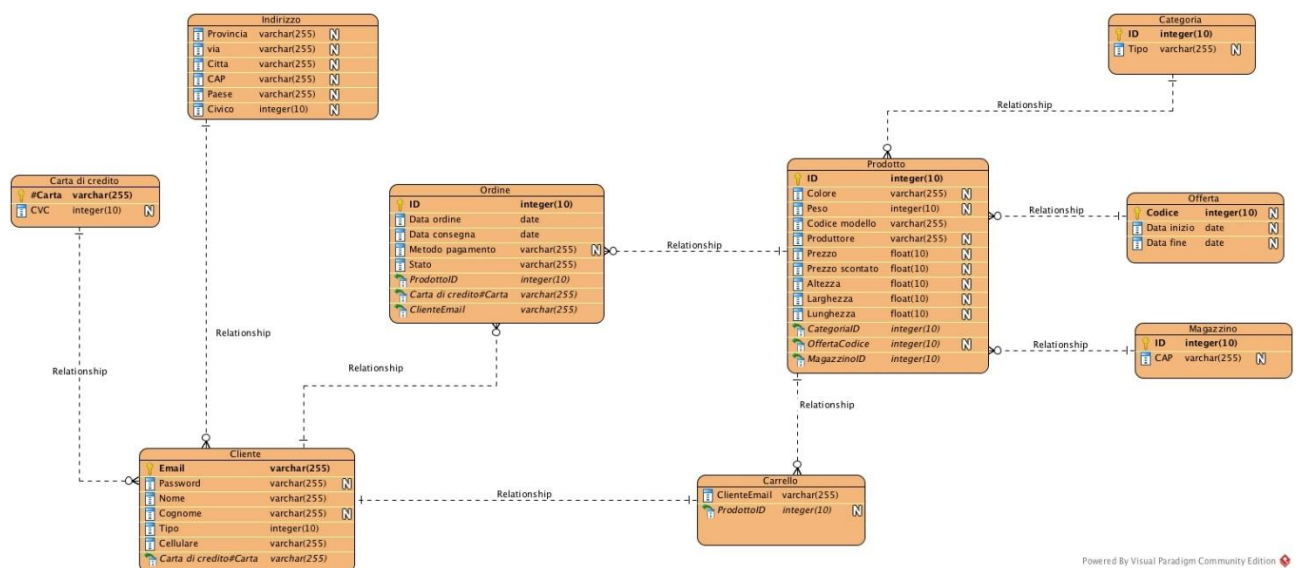
3.3 Software/hardware mapping

Dall'MVC precedente, abbia visto come l'architettura principale venga decomposta in 3 sottosistemi: model, control e view. Andiamo quindi ad analizzare ogni sottosistema, come verrà implementato e le tecnologie software/hardware utilizzate.

3.4 Gestione dei dati persistenti

L'infrastruttura di gestione dei dati necessaria alla memorizzazione dei dati persistenti che verrà utilizzata è un database Mysql. MySQL o Oracle MySQL è un Relational database management system, composto da un client a riga di comando e un server. Entrambi i software sono disponibili sia per sistemi Unix che per Window. I sistemi e i linguaggi di programmazione che supportano MySQL sono molto numerosi: ODBC, Java, Mono, .NET, PHP, Python e molti altri.

I dati verranno memorizzati secondo il seguente schema:



Powered By Visual Paradigm Community Edition

3.5 Access control and security

Il processo di e-commerce essenziale è necessario per il buon funzionamento e la gestione delle attività di e-commerce. Uno dei processi è il controllo degli accessi e la sicurezza. L'e-commerce deve stabilire una fiducia reciproca e un accesso sicuro tra le parti in una transazione di e-commerce, autenticando gli utenti, autorizzando l'accesso e rafforzando le funzionalità di sicurezza. Ad esempio, questi processi stabiliscono che un cliente e un sito di e-commerce sono quelli che dicono di essere attraverso nomi utente e password, chiavi di crittografia o certificati e firme digitali. Il sito di e-commerce deve quindi autorizzare l'accesso solo a quelle parti del sito che un singolo utente deve compiere le sue particolari transazioni. Di conseguenza, ai singoli individui verrà generalmente concesso l'accesso a tutte le risorse di un sito di e-commerce ad eccezione degli account di altre persone, dei dati aziendali riservati e delle aree di amministrazione dei webmaster. Un altro processo di sicurezza protegge la risorsa di un sito di e-commerce da minacce quali attacchi di hacker, furti di password o numeri di carte di credito e guasti di sistema. L'accesso alla nostra piattaforma, è reso sicuro da protocolli di rete verificati. Inoltre sono stati implementati meccanismi di verifica degli input per prevenire ipotetici errori.

3.6 Global software control

La seguente sezione descriverà l'implementazione del controllo software. Il flusso di controllo è distribuito all'interno del sistema di eleworld. Questo significa che non c'è istanza di controllo centrale, ma ogni servizio ha il proprio flusso di controllo. Utilizzando un modello MVC, la parte del controller è responsabile della logica di business. Questo componente ha la responsabilità di trasformare le interazioni dell'utente della View in azioni eseguite dal Model. Ma il Controller non rappresenta un semplice "ponte" tra View e Model. Realizzando la mappatura tra input dell'utente e processi eseguiti dal Model e selezionando la schermata della View richieste, il Controller implementa la logica di controllo dell'applicazione.

3.7 Boundary conditions

La maggioranza degli sforzi del system design sono indirizzati al comportamento steady-state. In questa fase bisogna però anche considerare la configurazione, l'avvio e la terminazione del sistema che si sta sviluppando.

- *Configurazione: ogni qual volta si accede al database, lo si fa tramite una classe dedicata (classe DAO). In questo modo si va a creare un oggetto persistente all'interno del sistema, in diretto contatto con il database.*
- *Inizializzazione: una volta creato l'oggetto persistente, viene inizializzato con i dati provenienti dal sistema, in particolar modo dai dati contenuti negli oggetti java.*
- *Terminazione: una volta utilizzati, gli oggetti terminano il loro compito, e rilasciano le risorse notificando ciò nel momento della terminazione della componente.*
- *Failure: il system design prevede fatal failure dovute da errori e bug. Ciò dà la possibilità di ripetere le varie operazioni (es. di rilascio), e salvaguardare i dati.*

4. Architettura sottosistema

4.1 Architettura Model

Il model definisce i dati e le operazioni che possono essere eseguite. Quindi definisce le regole di business per l'interazione con i dati, esponendo alla View ed al Controller rispettivamente le funzionalità per l'accesso e l'aggiornamento. Per lo sviluppo del Model verrà utilizzata una tipica tecnica di progettazione object oriented al fine di ottenere un componente software che astragga al meglio concetti importati dal mondo reale. In particolar modo verranno implementate diverse JavaBean, ovvero classi scritte in linguaggio di programmazione Java secondo una particolare convenzione. Sono utilizzate per incapsulare più oggetti in un oggetto singolo (il bean), cosicché tali oggetti possano essere passati come un singolo oggetto bean invece che come multipli oggetti individuali.

4.2 Architettura View

La logica di presentazione dei dati viene gestita solo e solamente dalla View. Ciò implica che questa deve fondamentalmente gestire la costruzione dell' interfaccia grafica (GUI) che rappresenta il mezzo mediante il quale gli utenti interagiranno con il sistema. Ogni GUI può essere costituita da schermate diverse che presentano più modi di interagire con i dati dell'applicazione. Nel caso della nostra architettura le View vengono implementate con JSP, le quali restituiscono GUI costituite solo da contenuti statici (HTML). Inoltre la View delega al Controller l'esecuzione dei processi richiesti dall'utente dopo averne catturato gli input e la scelta delle eventuali schermate da presentare.

4.3 Architettura Controller

Questo componente ha la responsabilità di trasformare le interazioni dell'utente della View in azioni eseguite dal Model. Ma il Controller non rappresenta un semplice "ponte" tra View e Model. Realizzando la mappatura tra input dell'utente e processi eseguiti dal Model e selezionando la schermate della View richieste, il Controller implementa la logica di controllo dell'applicazione. Nel nostro caso specifico, verranno implementate oggetti scritti in linguaggio Java che operano all'interno di un server web (es. Tomcat) oppure un server per applicazioni (es. GlassFish) permettendo la creazione di applicazione web.