

UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI MATEMATICA “TULLIO LEVI-CIVITA”

CORSO DI LAUREA IN INFORMATICA



## **Analisi e personalizzazione di portali B2B**

*Relatore*

Prof.ssa Ombretta Gaggi

*Laureando*

Beatrice Guerra

Anno accademico 2016-2017



*«Ciascuno chiama idee chiare quelle che hanno lo stesso grado di confusione delle sue.»*

Marcel Proust



Università degli Studi di Padova

## *Abstract*

Corso di Laurea in Informatica

Dipartimento di Matematica "Tullio Levi-Civita"

Laurea in Informatica

### **Analisi e personalizzazione di portali B2B**

di Beatrice Guerra

In questo lavoro viene analizzato lo sviluppo e l'utilizzo di portali *Business-to-Business*, effettuato durante lo stage svolto presso l'azienda Sanmarco Informatica S.p.A. (Grignano di Zocco - VI), all'interno della *business unit* 4words. La durata dello stage è stata di 312 ore.

L'obiettivo principale da raggiungere era la progettazione e realizzazione di personalizzazioni per portali web esistenti, sulla base di richieste effettuate dai clienti. Obiettivi secondari erano la conoscenza dell'impianto gestionale con cui il portale B2B interagisce, l'autonomia nel rapporto con il cliente stesso e la capacità di redigere analisi e stime per lo sviluppo delle nuove funzionalità richieste.

Per svolgere quanto pianificato, lo stage si è svolto per l'intera durata in affiancamento, sia in sede che presso clienti. È stata prodotta una sommatoria specifica per uno dei progetti seguiti, necessaria per avere una visione d'insieme del progetto, fortemente personalizzato. Sono seguite quindi alcune proposte implementative di nuove richieste, inizialmente svolte in collaborazione e successivamente sviluppate in autonomia.

Infine, è stata prodotta una analisi a posteriori di quanto visto, analizzando i vari aspetti caratterizzanti dei portali B2B, sia esternamente, sia internamente all'azienda ospitante.



## *Ringraziamenti*

The acknowledgments and the people to thank go here, don't forget to include your project advisor...





# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	L'azienda	1
1.1.1	4words	2
	Realizzazione siti web aziendali	2
	Realizzazione siti e-commerce B2C	2
	Realizzazione portali	2
	Realizzazione siti business B2B	3
1.2	Lo stage	3
1.2.1	Obiettivi	3
1.2.2	Ambiente di lavoro	3
1.3	Organizzazione del documento	4
<b>2</b>	<b>Il B2B</b>	<b>5</b>
2.1	Il B2B in generale	5
2.1.1	Il B2B come e-commerce	5
2.1.1.1	Funzionalità base	6
	L'ordine	6
	Il catalogo	6
	Caratteristiche generali degli e-commerce	6
	Filtri e ricerca	7
	Altre funzionalità	7
2.1.1.2	I fattori di successo del B2B	7
	Strategia di marketing	8
	Sito web	8
	Fattori globali	9
	Fattori interni	10
	Fattori esterni	10
2.2	Il B2B di 4words	11
2.2.1	Funzionalità principali	12
2.2.1.1	Abilitazioni e permessi	12
2.2.1.2	L'ordine	12
2.2.1.3	Parcheggio	13
2.2.1.4	Interrogazioni	13
2.2.1.5	Catalogo	13
2.2.2	Struttura del progetto	14
2.2.2.1	Vantaggi e svantaggi	16
	Vantaggi	16
	Svantaggi	16
2.2.3	Versionamento	17
2.2.3.1	Vantaggi e svantaggi	18
2.2.4	Tecnologie utilizzate	20
2.2.4.1	Ambiente di sviluppo	20

2.2.4.2	Back-end . . . . .	21
2.2.4.3	Front-end . . . . .	22
<b>3</b>	<b>Il progetto stage</b>	<b>25</b>
3.1	Primo progetto . . . . .	25
	Codice cliente . . . . .	26
3.2	Secondo progetto . . . . .	26
3.3	I due progetti a confronto . . . . .	26
<b>4</b>	<b>Conclusioni</b>	<b>27</b>
	<b>Bibliografia</b>	<b>35</b>

## Elenco delle figure

2.2.1 Pattern MVC a livello architetturale . . . . .	15
2.2.2 Struttura del sistema di versionamento RTC . . . . .	17
2.2.3 Flusso di lavoro tipico di RTC . . . . .	18
2.2.4 Creazione dei <i>bean</i> tramite <i>dependency injection</i> . . . . .	21
3.1.1 Pagina di accesso al portale . . . . .	25



## **Elenco delle tabelle**



## Capitolo 1

# Introduzione

In questo capitolo introduttivo vengono presentati il contesto aziendale in cui si è svolto lo stage e le premesse dello stage stesso. Infine, viene data una panoramica sul contenuto del documento.

### 1.1 L'azienda

Sanmarco Informatica nasce negli anni '80 come [software house](#) specializzata nello sviluppo di applicazioni gestionali per aziende manifatturiere ed è oggi una leading company italiana nella progettazione e realizzazione di soluzioni a supporto della riorganizzazione di vari processi aziendali e professionali. L'ambizione e la volontà di rinnovarsi hanno permesso all'azienda di evolversi attraverso esperienze e scelte imprenditoriali di successo, che individuano nella specializzazione del proprio capitale umano l'elemento centrale. L'azienda, partner di [International Business Machines Corporation \(IBM\)](#) Italia, cresce grazie all'impegno di 320 persone fra dipendenti e collaboratori, 13 distributori e 4 sedi – Grisignano di Zocco (VI), Reggio Emilia (RE), Tavagnacco (UD), Vimercate (MB). Le business units Sanmarco Informatica attive sono 5:

- **Jgalileo**: soluzione [Enterprise Resource Planning \(ERP\)](#) per diverse tipologie di aziende manifatturiere. Jgalileo costituisce il prodotto di punta dell'azienda, diventando sempre più all'avanguardia grazie all'impegno di oltre 80 consulenti del centro di ricerca e sviluppo software. Questa soluzione gestionale copre le esigenze dell'intero processo aziendale attraverso eccellenze applicative completamente integrate ed è progettato su misura per interpretare al meglio le tipicità di ogni mercato.
- **4words**: web apps marketing solutions. La web agency sviluppa app [iOS](#) e [Android](#) per attività di prevendita, cataloghi prodotti, raccolta ordini, assistenza tecnica, al fine di offrire strumenti sempre più efficaci per la gestione di processi aziendali ERP in mobilità. 4words guida anche il cliente all'utilizzo della piattaforma web a supporto di strategie commerciali [B2B](#) e [B2B](#).
- **NextBI**: analizza i dati dei clienti per ottimizzare le strategie di marketing. NextBI vanta infatti grande esperienza in progetti di [CI \(Customer Intelligence\)](#), [BI \(Business Intelligence\)](#) e [CPM \(Corporate Performance Management\)](#), fornendo moduli già pronti e trasversali a tutti i contesti aziendali.
- **Discovery Quality**: questa suite per la gestione della Qualità, leader di mercato, garantisce al management tutti gli elementi necessari per gestire e misurare

la Qualità e le prestazioni, gestendo efficacemente le istanze interne ed esterne di natura etica e sociale.

- **SMItech:** è il Team dedicato alla Tecnologia. L'obiettivo è aiutare i clienti a migliorare sicurezza ed efficienza informatica, attraverso la realizzazione di progetti di infrastruttura IT e lo sviluppo di servizi gestiti di [Cybersecurity](#).

Sanmarco Informatica è la prima ed unica azienda italiana entrata a far parte dell'[Open Power Foundation IBM](#); a gennaio 2016 l'azienda ha ricevuto il riconoscimento internazionale [Beacon Award](#) come finalisti a livello mondiale fra le aziende d'eccellenza che propongono soluzioni tecnologiche innovative in combinazione con il sistema [Power](#) di IBM.

### 1.1.1 4words

4words è una web agency che sviluppa siti web per il business, curandone la realizzazione (web design e criteri di usabilità) e le attività di web – marketing (posizionamento [SEO](#), campagne [PPC \(Pay Per Click\)](#) / AdWords, social media marketing, annunci AdSense, campagne video). 4words realizza inoltre applicazioni per tablet relative al catalogo prodotti, orientate alla vendita, integrabili con siti B2B e B2C e capaci di creare un sistema integrato tra siti e software gestionale Erp. La capacità di sviluppare applicazioni mobile integrabili con Sistemi Gestionali Erp rende 4words partner ideale per [PMI \(Piccole e Medie Imprese\)](#) e grandi aziende che vogliano fare un salto tecnologico significativo atto a cogliere potenzialità e vantaggi competitivi del web 2.0 e della connettività mobile.

**Realizzazione siti web aziendali** 4words cura la realizzazione e lo sviluppo di siti web aziendali, siti vetrina e catalogo prodotti al fine di far conoscere le realtà commerciali italiane su Internet. La realizzazione di un sito inizia dalla fase di pianificazione del progetto e della strategia da adottare per renderlo user – friendly e facilmente rintracciabile dai motori di ricerca. Successivamente, si passa alla cura della grafica e del design delle pagine. 4words è molto attenta a sviluppare siti web accattivanti e dai contenuti interessanti, al fine di migliorare il business e soddisfare i bisogni del cliente.

**Realizzazione siti e-commerce B2C** 4words si occupa della creazione di siti e-commerce per il tuo business realizzando shop online per la vendita di prodotti, basati su ampi database e configuratori semplici da utilizzare e ottimi per visionare e confrontare tutti gli articoli presenti sul sito. Il nostro team cura la realizzazione di e-commerce particolarmente efficaci sia dal punto di vista grafico sia dell'usabilità, affinché l'utente possa consultare rapidamente i prodotti e giungere successivamente al carrello, dove potrà concludere con semplicità l'acquisto.

**Realizzazione portali** 4words sviluppa portali B2C atti a contenere grandi quantità di informazioni e ideali per la raccolta dati relativi all'azienda (notizie su blog e forum, documenti consultabili e contenuti scaricabili dall'utente). I portali si rivolgono a tutti gli [Stakeholder](#), siano essi fornitori, dipendenti o clienti dell'azienda e sono molto utili per i processi decisionali interni all'impresa.



**Realizzazione siti business B2B** L'azienda crea e sviluppa siti business B2B, ideali per chi vuole far conoscere la propria azienda su Internet e sviluppare la sua reputazione, grazie a una corretta visibilità sul web. L'idea è quella di rendere visibili sia le PMI sia le grandi imprese, per realizzare nuove opportunità di business ed essere competitivi sul mercato. 4words adotta le migliori strategie affinché un'azienda sia facilmente rintracciabile sui motori di ricerca e sviluppi fedeltà nel cliente. Un cliente soddisfatto è un cliente che ritorna.

## 1.2 Lo stage

Lo stage prevedeva l'inserimento nel gruppo web 4words attraverso una prima parte di formazione tramite lezioni frontali e una seconda e più consistente parte in affiancamento, per applicare quanto fino ad ora imparato nel reale contesto aziendale. La durata dello stage è stata di 312 ore, suddivise in 8 settimane. Di queste, 72 ore sono state dedicate alla formazione, e nelle restanti 240 sono state svolte attività di analisi, progettazione e realizzazione di personalizzazioni per i portali web esistenti.

### 1.2.1 Obiettivi

Gli obiettivi pianificati erano i seguenti:

<b>Obbligatorî</b>
Progettazione dei portali Web per la raccolta degli ordini
Realizzazione software Java
Interazione con database SQL
Realizzazione front-end HTML, CSS, JavaScript
<b>Desiderabili</b>
Conoscenza dell'impianto commerciale del gestionale
Conoscenza dell'impianto amministrativo del gestionale
Autonomia della gestione con il cliente per raccolta nuove richieste
<b>Opzionali</b>
Analisi e stima nuove richieste clienti

### 1.2.2 Ambiente di lavoro

Lo stage si è svolto principalmente presso la sede dell'azienda, a Grisignano di Zocco (VI), con qualche uscita presso clienti. All'inizio dello stage mi è stata assegnata la strumentazione necessaria per poter svolgere il lavoro; mi è quindi stato affidato un portatile Lenovo ThinkPad L530, utilizzabile anche nel caso di spostamenti. Nel portatile erano già preinstallati i software di sviluppo utilizzati nell'azienda, tra cui l'[IDE Eclipse](#).

### 1.3 Organizzazione del documento

Il documento è così strutturato:

- Il **capitolo 2** presenta una analisi del B2B, a livello generale e nel contesto aziendale, analizzando la struttura del progetto, le tecnologie utilizzate, le funzionalità offerte, ed effettuando un confronto con altri contesti presenti nel mercato.
- Il **capitolo 3** descrive quanto svolto durante lo stage nella pratica. Vengono quindi presentati i due progetti interessati, prima a livello generale e poi nello specifico rispetto alle personalizzazioni effettuate.
- Nel **capitolo 4** vengono tratte alcune conclusioni sullo stage svolto, illustrando gli obiettivi raggiunti e le aspettative raccolte al termine dello stage stesso.

## Capitolo 2

# Il B2B

In questo capitolo viene descritto e analizzato il portale B2B, sia a livello generale che nello specifico del contesto aziendale in cui si è svolto lo stage. Vengono analizzate le funzionalità, i fattori di successo e le tecnologie utilizzate, presentandone vantaggi e svantaggi.

### 2.1 Il B2B in generale

Con B2B si intende un modello per la vendita di prodotti e servizi ad altre aziende. Un portale B2B può essere considerato come un'impresa di supporto che offre ciò di cui le altre aziende necessitano per vincere la concorrenza. Il B2B, infatti, offre prodotti grezzi, servizi o articoli in stock con prezzi molto vantaggiosi rispetto alla usuale vendita al pubblico, in quanto venduti il più delle volte direttamente dalla compagnia che li produce. Al contrario, nei portali B2C destinati al cliente singolo, il prodotto finito viene venduto con un prezzo che rispecchia tutte le piccole transazioni avvenute per la composizione dello stesso: l'azienda A che produce automobili ha dovuto acquistare i bulloni dall'azienda B, le vernici dall'azienda C e il vetro dei finestrini dall'azienda D. Tutti questi acquisti da terzi aumentano di fatto il costo dell'auto finale, che dovrà di per sé garantire la copertura delle spese ed un guadagno per il venditore.

Come qualsiasi altra attività commerciale, il modello B2B richiede una attenta pianificazione. Esso tipicamente fa affidamento su un rapporto solido che il team di vendita instaura con il cliente, mentre quanto inerente la promozione commerciale può includere pubblicità in riviste commerciali, *convention* e conferenze, *marketing* digitale (pubblicità online, tecniche SEO, newsletter) ed altre tecniche di sensibilizzazione tradizionali.

#### 2.1.1 Il B2B come e-commerce

Con la diffusione del web e la rivoluzione digitale, è emerso un nuovo settore commerciale definito B2B. Tramite portali online, le compagnie hanno iniziato a vendere direttamente ad altre aziende, così come a condividere i dati e informazioni riguardo ai prodotti ed ai servizi in modo facile e rapido. Un portale web è infatti sempre raggiungibile e permette una diffusione immediata delle informazioni.

Parlando di B2B possono essere individuate tre categorie principali:

**E-commerce** Molte aziende necessitano di raggiungere specificatamente altre aziende e i loro impiegati. Il sito web può rappresentare il collegamento diretto con

una rete esclusiva riservata ai clienti. Le aziende possono inoltre vendere direttamente dal sito prodotti che di fatto non richiedono di essere visti dal vivo o provati prima dell'acquisto. Un esempio sono i B2B che vendono a loro volta B2B, componibili direttamente dal cliente tramite editor, *template* e vari plugin per le più svariate funzionalità.

**Fornitura e offerta** Questi siti sono di solito destinati ad un mercato di nicchia. Un agente può acquistare materiale dal fornitore, richiedere una proposta di vendita ed anche effettuare offerte per comprare ad un prezzo specifico. È in questa categoria che ricade il B2B 4words.

**Informativi** La categoria finale è per i siti informativi, che forniscono informazioni specializzate per determinati settori. Questi portali sono spesso usati come siti organizzativi per gli standard commerciali e industriali.

### 2.1.1.1 Funzionalità base

Per essere un buon portale, un B2B deve includere alcune funzionalità fondamentali che ne determinano il successo. Queste comprendono ad esempio la gestione dell'ordine, la visualizzazione dei prodotti e la gestione dei clienti. Vi sono poi aspetti legati all'usabilità, come la semplicità nella creazione dell'ordine e l'efficienza nella ricerca di elementi come prodotti o clienti, ed altri inerenti le tecniche SEO.

**L'ordine** L'ordine è l'elemento alla base del B2B. Lo scopo primario di gran parte dei portali B2B è semplificare i processi interaziendali di fornitura e acquisto. L'ordine è anche l'entità più complessa che il B2B gestisce: vi sono infatti moltissimi fattori che dipendono non solo dall'azienda che ne dispone, ma anche dalle leggi statali e continentali in vigore, che possono cambiare anche frequentemente. Si pensi ad esempio all'[IVA](#), una tassa applicata solo in alcuni Paesi, con regole e percentuali differenti. Un buon B2B deve essere in grado, soprattutto se destinato ad un mercato internazionale, o comunque in espansione, di gestire tutte queste variabili, a seconda di dove sta chi vende e chi compra. Non a caso un'azienda per la gestione commerciale-amministrativa degli ordini necessita e si avvale di software gestionali, creati appositamente per questo scopo. Il B2B deve quindi essere in grado di riportare le logiche del gestionale nel web, rispettando le convenzioni di quest'ultimo e proponendo delle operazioni "guidate", in modo tale che il sistema sia utilizzabile anche da chi di gestionale ne sa poco o nulla.

**Il catalogo** I prodotti devono essere reperibili ed inseribili in un ordine. Il loro dettaglio deve essere chiaro: un prodotto è sempre caratterizzato da un codice che lo identifica in modo univoco e che viene spesso utilizzato per acquisti "rapidi", soprattutto quando vengono effettuati ordini ripetuti (un'azienda che utilizza per tutti i macchinari un certo bullone inserirà nell'ordine direttamente il codice "BX123Y", piuttosto che ricercare il prodotto tramite la stringa "Bullone con testa esagonale m8").

**Caratteristiche generali degli e-commerce** Essendo un portale destinato agli acquisti, il B2B deve avere le funzionalità base di questa categoria: un carrello per poter raggruppare i prodotti che si vogliono comprare e sapere in anticipo il prezzo

totale; un pannello di controllo per il cliente (la *dashboard*) per tenere traccia degli ultimi movimenti e del loro stato; l'accesso al profilo per la modifica di informazioni personali; la possibilità di effettuare il login, fondamentale per controllare le attività svolte dagli utenti.

**Filtri e ricerca** La ricerca di elementi all'interno del portale è una funzionalità che non può mancare, soprattutto con un numero di elementi molto elevato, come spesso avviene nei B2B. Allo stesso modo applicare dei filtri riduce i tempi che l'utente impiega per ottenere ciò che vuole, aumentando di fatto la sua soddisfazione.

**Altre funzionalità** Altre funzionalità che si possono definire *must-have* sono:

- configurazione dell'ordine;
- informazioni sulla disponibilità e sulla consegna;
- prezzo del prodotto pensato sul cliente;
- sconti e promozioni;
- pagamenti sicuri ed in varie modalità;
- *Search Engine Optimization*;

È necessario inoltre che siano supportati tutti i *browser* per le versioni più recenti e che le performance del sistema siano buone, sia in termini di caricamento delle pagine, sia per quanto riguarda i tempi di esecuzione di *query* di ricerca nel database.

#### 2.1.1.2 I fattori di successo del B2B

Il commercio *business-to-business* è sempre stato uno dei vantaggi principali per la predominanza delle aziende nella competizione internazionale. L'avvento di Internet e le nuove modalità di commercio e comunicazione hanno però modificato profondamente le dinamiche e i processi tradizionali per tale mercato. Da alcune ricerche sono quindi emersi i fattori critici che un'azienda deve considerare per avere successo in questo settore: 21 elementi, suddivisi in 5 categorie (strategia di marketing, sito web, dimensione globale, fattori interni e fattori esterni). Per individuarli sono state usate alcune tecniche specifiche, come

- scansione ambientale;
- analisi della struttura industriale;
- opinione di esperti del settore;
- analisi della concorrenza;
- *best practice*;
- valutazione interna;
- fattori di intuizione;
- analisi del profitto di strategie commerciali.

Per quanto invece riguarda i fattori proposti, la forza vendita gioca un ruolo centrale nello sviluppo delle strategie di mercato se viene fornita una formazione appropriata. Il coinvolgimento di fornitori e clienti, la cultura del web e l'utilizzo di entrambi i mezzi (tradizionale ed online) sono altri fattori importanti, così come la sicurezza, la fiducia e la confidenza tra il venditore ed il potenziale cliente.

### Strategia di marketing

**Supporto e impegno manageriale** È richiesta la conoscenza delle potenzialità di Internet da parte del manager, che ha anche il compito di diffonderle in modo proattivo all'interno dell'azienda. L'impegno preso verso il commercio nel web aiuta a promuovere il suo sviluppo anche in altre aziende, ma richiede comunque supporto finanziario: c'è un'importante correlazione tra l'investimento effettuato ed il guadagno ottenuto. Per questo il coinvolgimento del livello amministrativo gioca il ruolo più critico.

**Obiettivi strategici** Il successo dello sviluppo del web B2B dipende da quanto chiaramente sono definiti gli obiettivi strategici per una organizzazione.

**Integrazione tra Internet e la strategia di marketing** I responsabili marketing web di successo sono quelli che, oltre a costruire un sistema in grado di integrarsi con le applicazioni esistenti, sappiano offrire la formazione necessaria per il suo utilizzo. Sebbene Internet offra molti dei servizi comuni, esso non rimpiazzerà i mezzi tradizionali: i clienti che comprano online continuano comunque a comprare attraverso altri mezzi. Pertanto, le aziende devono considerare il web-marketing come un complemento, piuttosto che un rimpiazzo. Molti clienti infatti preferiscono valutare un prodotto online e poi procedere con l'acquisto in altri modi, di persona o al telefono ad esempio.

**Definizione del target** Definire chi tra clienti esterni, fornitori, venditori, rivenditori ed altri *business partner* costituisce il target di vendita è una operazione primaria, in quanto determina come i canali, gli strumenti e il target stesso vadano utilizzati e coinvolti.

### Sito web

**Design** Un sito web ben progettato e con un bel design è il biglietto da visita dell'azienda nel web. La creazione di un portale richiede però un continuo sforzo per il suo aggiornamento e mantenimento, per continuare a soddisfare le aspettative degli utenti in base alle tendenze. Un ulteriore fattore è il contenuto, che deve essere di valore, accurato ed aggiornato, sia per attrarre nuovi clienti da ogni dove, sia per incoraggiarli a ritornare; le informazioni devono essere chiare e consistenti, in quanto i clienti fanno sempre una valutazione sulla loro utilità nel condurlo a prendere decisioni. La maggior parte degli utenti arriva sul web in cerca di informazioni, pertanto un portale che offre più dettagli sull'azienda e sui prodotti avrà sicuramente più successo. D'altro canto, le performance per l'accesso a tali informazioni sono un

dato altrettanto importante: la progettazione del catalogo dei prodotti va sviluppata ancor prima dello stesso portale, in quanto determina la velocità di esecuzione delle *query* di ricerca per migliaia di prodotti.

Vanno quindi considerate tutte le regole base di usabilità, tra cui la semplificazione della navigazione con [breadcrumb](#) e [tag](#), l'interazione e la reattività verso i [feedback](#) degli utenti, lo scambio di informazioni tra utenti e l'integrazione del web con altri canali di marketing. "Il sito web non deve servire solamente come interfaccia di raccolta ordini ma ha anche un alto valore aggiunto per il contenuto informativo".<sup>1</sup>

**Promozione (offline ed online)** La promozione è importante per due motivi: il proprio sito web deve essere riconoscibile rispetto a quello della competizione. Questo richiede un'alta accessibilità, cosicché sia il sito che la sua pubblicità possano raggiungere il maggior numero di utenti possibile. Secondo, la promozione di un sito web richiede le conoscenze tecniche di un esperto che sappia come l'utente medio trova usualmente i contenuti su Internet.

### Fattori globali

**Comprensione dell'ambiente esterno** Un sito dovrebbe rispettare il sistema degli stati in cui è utilizzato. Vanno quindi studiati gli ambienti, incluse le regolazioni del commercio e le modalità di spedizione, per comprendere i vantaggi dei prodotti e dei servizi locali. Il mercato internazionale richiede che vengano effettuate molte considerazioni gestionali e di pianificazione, tra cui lo standard dei prodotti locali, i prezzi di mercato, i fattori competitivi, la valuta, i problemi delle modalità di pagamento, l'assistenza, i servizi offerti ai clienti e considerazioni riguardanti le leggi locali.

**Risorse** Per essere pronti all'incremento delle vendite che il portale web può portare, sono necessarie risorse che non tutte le aziende potrebbero avere. Alcuni esempi sono la possibilità di ricevere ordini 24 ore su 24, un servizio clienti efficiente e l'esperienza per la gestione delle spedizioni internazionali.

**Multilingua** Uno dei problemi principali per la comunicazione a livello globale è la lingua. Per le compagnie che vogliono utilizzare il web a livello internazionale, la lingua è la sfida da superare. Per il commercio internazionale è necessario fornire una traduzione per almeno le lingue base, in modo tale da ridurre le difficoltà di comprensione per i lettori non madre-lingua.

**Considerazioni culturali** Così come la lingua, anche gli altri fattori culturali devono essere riportati nel portale web. Inoltre, fornire informazioni che possano risultare interessanti da una varietà di persone con bisogni e gusti diversi può incoraggiare l'incontro tra le differenti nazioni e culture. Va considerato che anche gli elementi non testuali possono causare incomprensioni culturali, come per esempio l'uso

---

<sup>1</sup>Riyad Eid, Myfanwy Trueman e Abdel Moneim Ahmed. «A cross-industry review of B2B critical success factors». In: *Internet Research* 12.2 (2002).

dei colori o dei simboli delle icone (il colore bianco, che in gran parte del mondo è neutrale o addirittura elegante, in alcuni Paesi dell'Asia significa morte).

**Consegna internazionale** Progettare un sistema logistico in grado di effettuare consegne in più nazioni efficientemente è necessario se si vuole vendere in più stati e favorisce la scalabilità per le medie aziende in espansione. Come minimo, quindi, il portale B2B dovrebbe indicare i costi e le tempistiche di spedizione verso tutti i Paesi in cui è disponibile la consegna.

### Fattori interni

**Infrastruttura tecnologica** Affinché tutto il sistema di cui si è parlato fino ad ora funzioni correttamente, è necessario che l'infrastruttura tecnologica alla base offra buone prestazioni. Questo non è comunque l'unico aspetto da considerare, in quanto Internet per essere uno strumento utile ha bisogno che i suoi utilizzatori abbiano familiarità con il PC e sappiano apprezzare i benefici e le potenzialità che esso offre.

**Cultura interna** Un'azienda deve comprendere e abbracciare i nuovi valori, i processi di gestione e gli stili di comunicazione che i nuovi modi di fare marketing creano. Iniziare a commerciare via web è come trasferirsi in un nuovo stato: la chiave del successo sta nel comprendere, apprezzare e onorare la cultura ed i protocolli di quel Paese.

**Forza vendita** La forza vendita ha un ruolo centrale nel successo di un portale B2B, in quanto in grado di migliorarne l'espressività, aiutando il cliente ad avvicinarsi al portale e integrando nuovi strumenti di gestione dell'informazione. Le conoscenze degli addetti al marketing, inoltre, rimangono fondamentali, dato che un sito per quanto tecnologicamente avanzato non può aver successo se non rispetta le aspettative del cliente.

**Formazione** Una adeguata formazione risulta importante per mantenere aggiornati gli utenti, siano essi interni o esterni, sulle nuove funzionalità e potenzialità del sistema.

### Fattori esterni

**Fiducia** Più l'utente è cosciente di quello che sta facendo nel web, più vorrà delle garanzie. La fiducia verso un sito web sta quindi diventando la chiave che determinerà il successo o il fallimento di molte compagnie. La fiducia è ormai più importante nel mondo virtuale che in quello reale, poiché le parti coinvolte in operazioni come l'ordine non sono più nello stesso luogo e di certo non possono dipendere su variabili come la prossimità fisica, la stretta di mano o i segnali del corpo.



**Sicurezza** Una delle preoccupazioni più diffuse nel web è la sicurezza delle transazioni finanziarie, tanto che alcuni preferiscono guardare i prodotti online, per poi utilizzare i metodi tradizionali per effettuare i pagamenti.

**Relazioni** I continui cambiamenti che avvengono nel mondo dell'informatica hanno determinato un cambiamento nelle relazioni: esse si basano sempre più sull'informazione che può essere trasmessa tra aziende, piuttosto che sulle modalità con cui questa transazione avviene, come prevede invece la visione tradizionale. Svolgono un ruolo fondamentale quindi i contenuti, le tecnologie e le strategie di marketing.

**Infrastruttura di rete** Uno dei fattori esterni che non dipendono direttamente dall'azienda è l'infrastruttura di rete: per sfruttare al massimo i vantaggi che il mercato virtuale offre, è necessaria un'infrastruttura in grado di supportare le novità che il settore propone.

**Coinvolgimento del cliente** Come ultimo punto, ma non meno importante, è il coinvolgimento del cliente in questo tipo di attività. Le aziende dovrebbero impegnarsi a motivare i clienti ad effettuare il passaggio all'ambiente online, oltre che a predisporre un sistema interno in grado di rispondere rapidamente alle richieste dei clienti. Il modello ottimale che soddisfa a pieno i clienti si basa infatti su una infrastruttura *e-business* con quattro caratteristiche: è facile da usare, ha molte funzionalità, è affidabile ed offre prestazioni elevate.

## 2.2 Il B2B di 4words

Il B2B di Sanmarco Informatica è un prodotto completo ed al contempo in continua evoluzione. Svolge tutte le funzionalità base che un buon B2B dovrebbe avere e si integra pienamente con gli altri servizi dell'azienda, in particolar modo con il gestionale, agevolando così attività quali l'inserimento e la gestione degli ordini, la gestione dei documenti e degli appuntamenti, la visualizzazione del catalogo, la configurazione di prodotti. Il B2B è fornito in varie versioni per adattarsi meglio alle esigenze dei clienti:

- versione standard (il più diffuso);
- versione con web-services;
- versione moda.

In questa relazione viene trattata solamente la versione standard, l'unica su cui si è incentrato lo stage, mentre verrà solamente presentata la versione con web-services.

Ogni B2B venduto è personalizzato, anche solo in minima parte, secondo le richieste del cliente. È raro dunque incontrare due portali identici, poiché per quanto essi possano assomigliarsi per forma e contenuti, vi sono sempre delle funzionalità da una parte che l'altra ha, per scelta del cliente, nascoste. Esistono versioni molto basilari, in cui viene personalizzato solamente l'aspetto o, ad esempio, il menu, ed altre in cui ritrovare lo standard è quasi un'impresa. Molto spesso l'utilizzo sempre più massivo del portale fa sì che sorgano proposte da parte dell'utilizzatore, che

vede nel B2B la soluzione per snellire alcune pratiche altrimenti richiedenti l'uso del gestionale.

### 2.2.1 Funzionalità principali

In generale, il portale è composto da moduli componibili, in modo tale da essere di per sé un prodotto funzionante, ma anche facilmente personalizzabile secondo le necessità del cliente. I moduli principali sono quello dell'ordine, delle interrogazioni, l'area amministrativa, l'area dei documenti e l'agenda. Tutte queste aree sono attivabili o disattivabili per ogni utente, a seconda del suo gruppo di appartenenza.

#### 2.2.1.1 Abilitazioni e permessi

Una delle funzionalità principali è infatti la caratterizzazione degli utenti per ruoli. Ad ogni ruolo può essere associata un'abilitazione differente, in modo tale da adattare l'esperienza di navigazione per ognuno di essi. Si considerino per esempio gli amministratori, gli agenti di vendita e i clienti: sono tre tipi diversi di utenza, ai quali vanno mostrate aree differenti. Gli amministratori possono avere il controllo su tutto, in particolar modo sulle configurazioni. Sono in grado quindi di creare, modificare ed eliminare utenti, bloccarli o sbloccarli, gestire gli ordini, creare, modificare e rimuovere abilitazioni con i relativi permessi, e molto altro. L'amministratore di fatto ha il controllo completo del B2B, può arrivare a qualsiasi area, anche quelle a più basso livello, come la creazione delle connessioni ai database. L'agente di vendita potrebbe invece poter solamente creare ordini, modificandoli fin tanto che non sono inviati, creare e gestire i propri clienti, senza vedere quelli di altri agenti, così come per gli appuntamenti. La navigazione dell'agente è quindi totalmente focalizzata sulla sua funzione all'interno dell'azienda e, come nel sistema tradizionale, egli non ha accesso a informazioni riguardanti suoi co-ruolo. Possono fare eccezione i capogruppo, ovvero agenti con un grado di controllo maggiore, che permette loro di controllare le attività dei subordinati. Il ruolo di cliente, infine, è generalmente quello con i permessi minori: il cliente di solito è un'altra azienda o organizzazione, che non deve quindi avere accesso ad informazioni riservate.

Le abilitazioni e i permessi sono di fatto del tutto personalizzabili senza l'ausilio di un tecnico 4words. Gli utenti sono inseribili, come visto, in gerarchie, che semplificano ancor di più la gestione da parte dell'amministratore, il quale in pochi passi è in grado di definire regole per molti utenti contemporaneamente. All'utilizzatore finale tutta questa serie di permessi è totalmente trasparente, in quanto non vede che esistono altre aree a lui inaccessibili.

#### 2.2.1.2 L'ordine

Come per ogni B2B, l'ordine rappresenta il fulcro del portale e l'entità più complessa. Per rendere la creazione dell'ordine completa e al tempo stesso *user-friendly*, essa è stata suddivisa in vari step che aiutano l'utente a inserire le informazioni richieste correttamente, presentandogli un numero ridotto di informazioni da fornire e, dove possibile, con aiuti, come ad esempio la scelta multipla (e dunque limitata). Tramite questo [wizard](#) è quindi possibile inviare ordini, ma anche iniziarli e lasciarli in sospeso per inviarli successivamente.

### 2.2.1.3 Parcheggio

Una volta inviato, l'ordine può essere inserito direttamente nel gestionale oppure ricevere dei blocchi (amministrativi o commerciali ad esempio), i quali determinano la necessità di approvazione da parte dell'ufficio relativo per il suo processamento. Questo blocco è detto parcheggio e, con tutte le personalizzazioni possibili in termini di regole e modalità, è un'area molto utile in quanto velocizza alcune operazioni standard. Un tipico esempio di utilizzo è per i clienti fuori fido: a questa categoria viene permesso di fare ordini solamente a certe condizioni, verificate personalmente da alcuni addetti. Ecco quindi che il B2B permette a tali persone di accedere direttamente a tutti gli ordini sottoposti a controllo, sbloccandoli o rifiutandoli con un semplice click, con la possibilità di notificare l'utente di quanto deciso via email. Questa procedura semplifica molto operazioni che fatte direttamente nel gestionale potrebbero risultare non solo più difficoltose, ma anche meno immediate, in quanto per accedere al B2B è sufficiente un tablet o un pc connesso ad internet, mentre il gestionale è solitamente disponibile solamente dentro all'azienda.

### 2.2.1.4 Interrogazioni

L'area delle interrogazioni è dedicata alla ricerca di schede e informazioni riguardanti i dati inseriti dall'utente. Questi dati sono suddivisi in varie categorie, così da permettere all'utente di trovare facilmente quello che cerca. In particolare, l'elenco dei clienti viene raccolto in un'unica videata, con tutte le loro informazioni. Anche per gli ordini esiste uno storico che ne indica i dettagli, dove è possibile applicare dei filtri, velocizzando molto il procedimento di ricerca.

### 2.2.1.5 Catalogo

Il catalogo prodotti nella sua versione standard rispetta i requisiti del modello B2B di successo. Esso è presentato in varie modalità, secondo le convenzioni del web, tra cui la visualizzazione a lista (con più dettagli per ogni articolo) e quella a griglia, in cui hanno un ruolo importante le immagini. Questa seconda modalità è di solito rivolta a chi con il proprio B2B vuole rispecchiare gli e-commerce più famosi, come Amazon o Ebay, piuttosto che a clienti che vendono prodotti tecnici di cui sono più importanti le caratteristiche della forma. Il catalogo B2B non solo permette di effettuare ordini secondo le abitudini degli utenti (il 66% degli utenti di Internet ha fatto acquisti online nel 2016<sup>2</sup>), ma ha anche la funzione di sostituire i cataloghi stampati che gli agenti di vendita propongono al cliente, in quanto è legato all'utente autenticato ed è una fonte sempre aggiornata. Ad ogni prodotto è possibile associare la disponibilità, aggiornata in tempo reale rispetto al gestionale, riducendo così significativamente il numero di ordini con prodotti non disponibili o non più in produzione. Per l'agente questa è una caratteristica importante, in quanto non è più necessario mettersi in contatto con la sede centrale per conoscere cambiamenti effettuati ai prodotti, anche in termini di prezzi e listini.

---

<sup>2</sup>E-commerce statistics for individuals. Eurostat. 2016. URL: [http://ec.europa.eu/eurostat/statistics-explained/index.php/E-commerce\\_statistics\\_for\\_individuals](http://ec.europa.eu/eurostat/statistics-explained/index.php/E-commerce_statistics_for_individuals).

### 2.2.2 Struttura del progetto

Passando agli aspetti implementativi, il B2B di 4words è strutturato gerarchicamente in vari progetti, con in testa alla gerarchia la versione base e al di sotto varie versioni dipendenti dalle soprstanti. In particolare la versione standard precedentemente citata è così formata:

- B2B Base: costituisce gran parte del back-end dell'applicazione. Contiene pressoché tutta la logica del portale e non ha alcun tipo di riferimento al front-end. Si occupa principalmente di gestire i dati del gestionale, riproducendoli sotto forma di oggetti.
- B2B Prime: è il nucleo del front-end. Include le librerie grafiche e predispone la totalità delle pagine navigabili nel portale. Si occupa anche di collegare il front-end alle classi del back-end, aggiungendo o aggiustando gli oggetti per renderli usufruibili dalla parte grafica.
- B2B R14: costituisce l'ultima grossa release di creazione del portale e rappresenta l'ultimo strato comune a tutti i progetti. È il vero e proprio applicativo ed in esso vengono definiti i denominatori comuni per creare uno standard stabile e duraturo, oltre che compatibile con i molti moduli disponibili per l'installazione.

Questi tre progetti formano quello che si è soliti definire lo standard del B2B. Esso rende l'installazione per il cliente semplice e immediata in quanto configura già di suo la maggior parte delle funzionalità. Da questi tre progetti si sviluppano poi una serie di moduli tra loro indipendenti, tra cui:

- B2B TCE: è il modulo che integra il B2B con il configuratore di prodotto offerto dall'azienda.
- B2B Rest Service: include le librerie necessarie per interrogare le [API](#) di servizi [REST](#), come ad esempio di [Magento](#) o [Prestashop](#). È il ponte tra il B2B e i vari [CMS](#) disponibili online e il suo scopo è proprio quello di agevolare l'interfacciamento tra questi due tipi di contenuto.
- B2B Rest Client: costituisce l'interfaccia grafica dell'integrazione B2B con i servizi web.

A partire da questi progetti vengono quindi creati i prodotti finali venduti ai clienti. Ogni cliente ha il suo progetto dedicato, in quanto sorgono in continuazione nuove richieste, sviluppate quindi in un ambiente indipendente, in modo tale da non creare conflitti con implementazioni contrarie di altri clienti.

Il B2B è strutturato secondo il pattern [MVC \(Model View Controller\)](#). Questo pattern è individuabile in più livelli:

- Nell'architettura ad alto livello, il *model* è composto dai componenti che contengono la logica dell'applicazione, in particolare da *manager*, *data provider* e *program provider*; la *view* è composta dal "codice JSF"; il controller è costituito dalla *FacesServlet*, una [servlet](#) che gestisce le richieste delle applicazioni web.
- La view del precedente livello rispecchia a sua volta il pattern, dove il *model* è costituito dalle entità, la *view* è composta dalle pagine [XHTML](#) (le [facelets](#)) e il controller è formato dai *backing bean*.

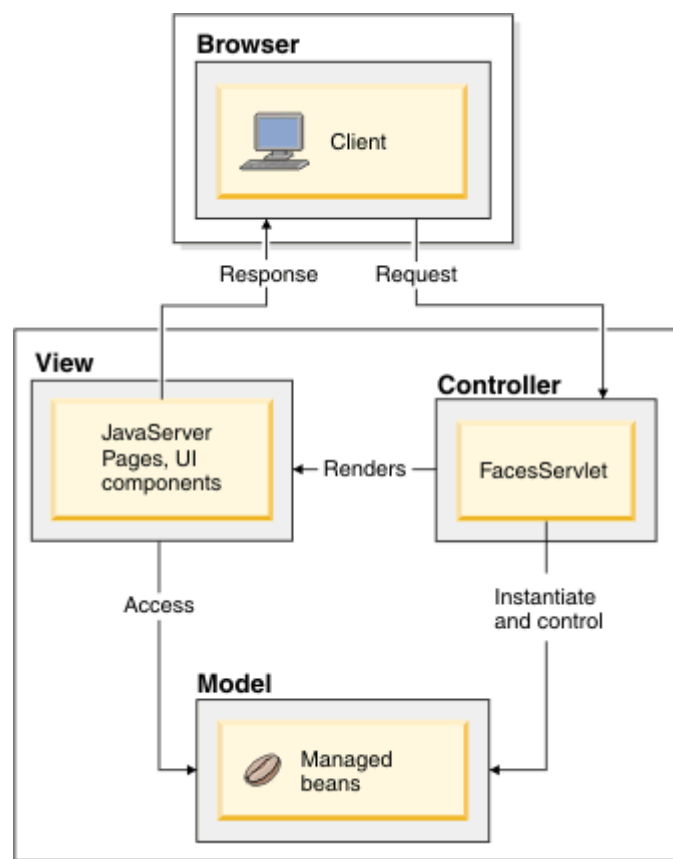


FIGURA 2.2.1: Pattern MVC a livello architetturale

L'architettura può quindi essere riassunta con lo schema M(MVC)C. Questa ricorsione si potrebbe protrarre anche al livello del client, aggiungendo quindi un ulteriore livello interno.

Nel B2B il *model* architetturale è stato suddiviso in vari *package*, citati precedentemente. I *manager* contengono la logica dell'applicazione e collegano la *view* ai *data provider*, i quali rappresentano il *data layer* nel pattern architetturale DAO ([Data Access Object](#)). Questi, inoltre, possono fare riferimento al loro interno ai *program provider*, i quali rappresentano l'interfaccia Java per i programmi RPG dell'AS/400. Secondo le norme di programmazione prestabilite per il progetto B2B, dunque, il flusso esecutivo per una operazione che richiede la chiamata ad un programma RPG è il seguente: la pagina fa riferimento ad un metodo del *backing bean*, che chiama il manager di competenza, il quale trasferisce la richiesta al *data provider*, che richiama il *program provider*. Il risultato della chiamata al programma torna quindi indietro attraverso *data provider*, *manager* e *backing bean*, che effettueranno eventuali controlli e operazioni necessari. Infine, se richiesto dall'applicativo, la pagina viene aggiornata e/o nuovamente renderizzata.

Quasi la totalità delle entità concepite nel B2B 4words ha un oggetto corrispettivo nei *backing bean*, nei *manager*, nei *data provider* e, a volte, nei *program provider*. Questa suddivisione permette di applicare il *single responsibility principle* (principio di singola responsabilità), secondo il quale ogni elemento deve avere una singola responsabilità, interamente incapsulata dall'elemento stesso. Ad esempio, l'entità *Order* è gestita rispettivamente nell'*OrderBck* per quanto riguarda il suo interfacciamento con le pagine JSF, nell'*OrderManager* per la logica e nell'*OrderDataProvider*

per l'accesso al database. Anche le personalizzazioni risultano così immediate, in quanto, per la gerarchia descritta precedentemente, è sufficiente derivare la classe che richiede la personalizzazione per usufruire di tutti i vantaggi che la struttura porta con sé, tra cui il polimorfismo.

Le pagine XHTML sono strutturate in modo tale da permettere il riutilizzo ove possibile, tramite template e componenti creati ad hoc per svolgere alcuni compiti specifici, come il calendario o il campo di testo per la valuta. Questo è reso possibile anche grazie all'utilizzo di JSF.

### 2.2.2.1 Vantaggi e svantaggi

La struttura appena descritta presenta alcuni vantaggi e svantaggi importanti.

**Vantaggi** Il primo vantaggio è che la suddivisione in più progetti permette che essi siano gestiti da un gruppo limitato di persone, riducendo il rischio di creare conflitti nelle modifiche contemporanee. Inoltre, ogni progetto cliente è indipendente dagli altri e dipende solo dai pacchetti di cui necessita: non vengono installate librerie al di fuori di quelle richieste per il corretto funzionamento, mantenendo quanto si va ad installare il più leggero possibile.

La separazione in più progetti permette anche una distribuzione dei ruoli all'interno del gruppo di sviluppo: l'implementazione di nuove funzionalità generali e solitamente di grosso impatto è affidata al gruppo standard; la personalizzazione delle funzionalità esistenti e lo sviluppo di funzionalità specifiche o non comuni, che quindi risulterebbero utili a pochi, è lasciato ai sistemisti con a carico il cliente. Nel caso in cui sorgano anomalie o problematiche negli ambienti dei clienti riguardo a moduli standard, interviene l'assistenza, la quale si occupa di individuare e correggere gli errori cosicché possa essere rilasciato un aggiornamento risolutivo comune per tutti i clienti.

La struttura gerarchica fortemente basata sull'ereditarietà facilita anche chi si addentra per la prima volta in un progetto così ampio nell'individuare il flusso d'esecuzione. I passaggi e le chiamate tra metodi seguono infatti sempre la stessa linea: pagina, *backing bean*, *manager*, *data provider*, *program provider* e ritorno.

**Svantaggi** Uno degli svantaggi principali nell'utilizzo di questa struttura è che con aggiornamenti del pacchetto standard importanti, laddove il cliente ha un portale molto personalizzato, c'è il rischio che le personalizzazioni si invalidino, richiedendo di fatto un lungo lavoro di "riporto" delle classi personalizzate per mantenerle compatibili con la nuova versione standard. Vi sono di fatto due possibilità: aggiornare l'intero pacchetto standard e modificare di conseguenza la parte personalizzata o procedere al contrario, aggiornando solamente ciò che aggiunge funzionalità o che corregge comportamenti anomali, senza modificare i metodi standard richiamati dalle classi personalizzate. Nel primo caso, come detto, il lavoro può risultare molto lungo, mentre nel secondo caso si rischia di arrivare ad uno stato di inconsistenza, dove più classi richiedono che uno stesso metodo svolga operazioni diverse, dovendo quindi comunque effettuare modifiche a svariate personalizzazioni. Tutto questo prevede anche una lunga fase di test, ai quali non viene mai dedicato il giusto tempo.

### 2.2.3 Versionamento

Lo strumento di versionamento utilizzato per tutti i progetti riguardanti il B2B è [RTC \(Rational Team Concert\)](#). Esso si basa su IBM Jazz, una piattaforma estendibile che aiuta i team ad integrare i task durante il ciclo di vita del software. RTC è costruito su una architettura client-server ed è integrabile con molti altri prodotti, tra cui [Git](#), [Jenkins](#) e [Maven](#). Per quanto riguarda la funzione di versionamento, RTC è così strutturato: sul server sono presenti gli *stream*, dei contenitori per uno o più progetti, accessibili solamente tramite invito. Agli utenti invitati vengono attribuiti uno o più ruoli, ad esempio *scrum master* o membro del team (solitamente lo sviluppatore), che determinano i permessi a loro associati.

RTC è strutturato secondo lo schema in figura 2.2.2.

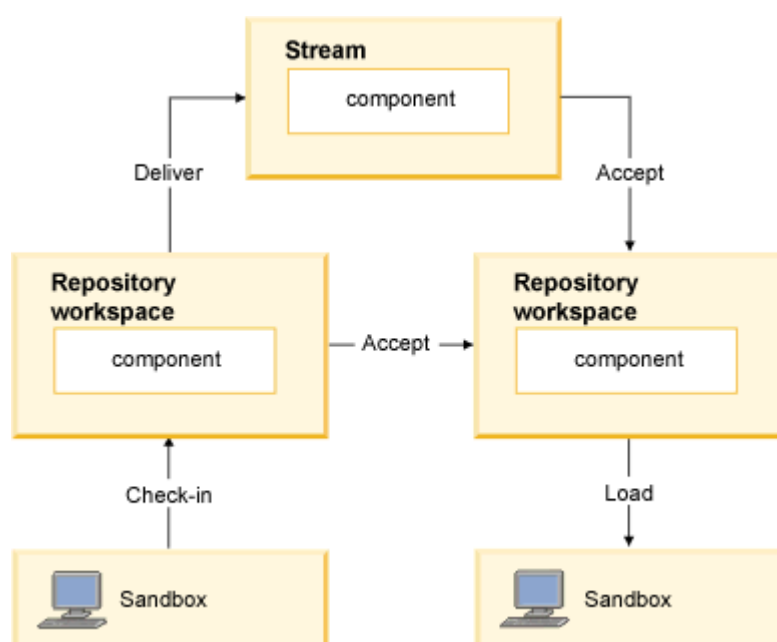


FIGURA 2.2.2: Struttura del sistema di versionamento RTC

Il sistema è centralizzato sullo *stream*, sul quale confluiscono tutte le modifiche e dal quale gli utenti scaricano i progetti. Vi è poi un livello intermedio, appartenente ai singoli utenti, che si trova sul server e permette agli utenti di salvare le proprie modifiche online, così da non correre il rischio di perderle a causa di malfunzionamenti hardware. Questo strato è chiamato *repository workspace*. Infine, il livello più basso è quello che risiede in locale, nel PC in cui si lavora. Le operazioni tra livelli hanno nomi specifici: il *check-in* è il trasferimento delle modifiche dal *workspace* locale a quello remoto; con *deliver* si intende il trasferimento delle modifiche dal *workspace* remoto allo *stream*. Con questa operazione le modifiche diventano effettive e scaricabili da tutti gli utenti, i quali devono accettarle (*accept*) per trasferirle al proprio *workspace* remoto e quindi riportarle in ambiente locale tramite l'operazione *load*.

Le funzionalità principali di RTC utilizzate per lo sviluppo del B2B sono:

- *work item*: sono il meccanismo sui cui si basa RTC per tracciare e coordinare i *task* di sviluppo e il flusso di lavoro. Sono il collegamento tra varie funzionalità di RTC, come le *build* e i *change set* (l'insieme delle modifiche associate, appunto, al *work item*). I *work item* possono assumere una categoria diversa a seconda

del loro contesto, come ad esempio “implementazione” o “bug”, per distinguere se le modifiche rilasciate costituiscono una nuova implementazione o risolvono una anomalia presente nel codice.

- **Controllo dei sorgenti:** è un sistema di controllo di versione costruito sulla piattaforma Jazz. Offre supporto per lo sviluppo parallelo con modello agile, integrando anche un sistema di gestione degli errori.
- **Pianificazione:** il componente per la pianificazione fornisce uno strumento di assistenza nell’organizzazione e svolgimento di progetti agile e non. Per lo sviluppo agile permette di salvare lo stato dei lavori ad una certa *release* o ad uno *sprint* e di tenere traccia del progresso durante le iterazioni, bilanciando il lavoro tra gli sviluppatori. Questo modello viene utilizzato soprattutto per lo sviluppo dei progetti standard, in quanto le personalizzazioni sono solitamente affidate ad una o due persone.

RTC è usato tramite la sua integrazione per Eclipse, l’IDE di sviluppo del B2B. Un tipico esempio di utilizzo di RTC è presentato in figura 2.2.3.

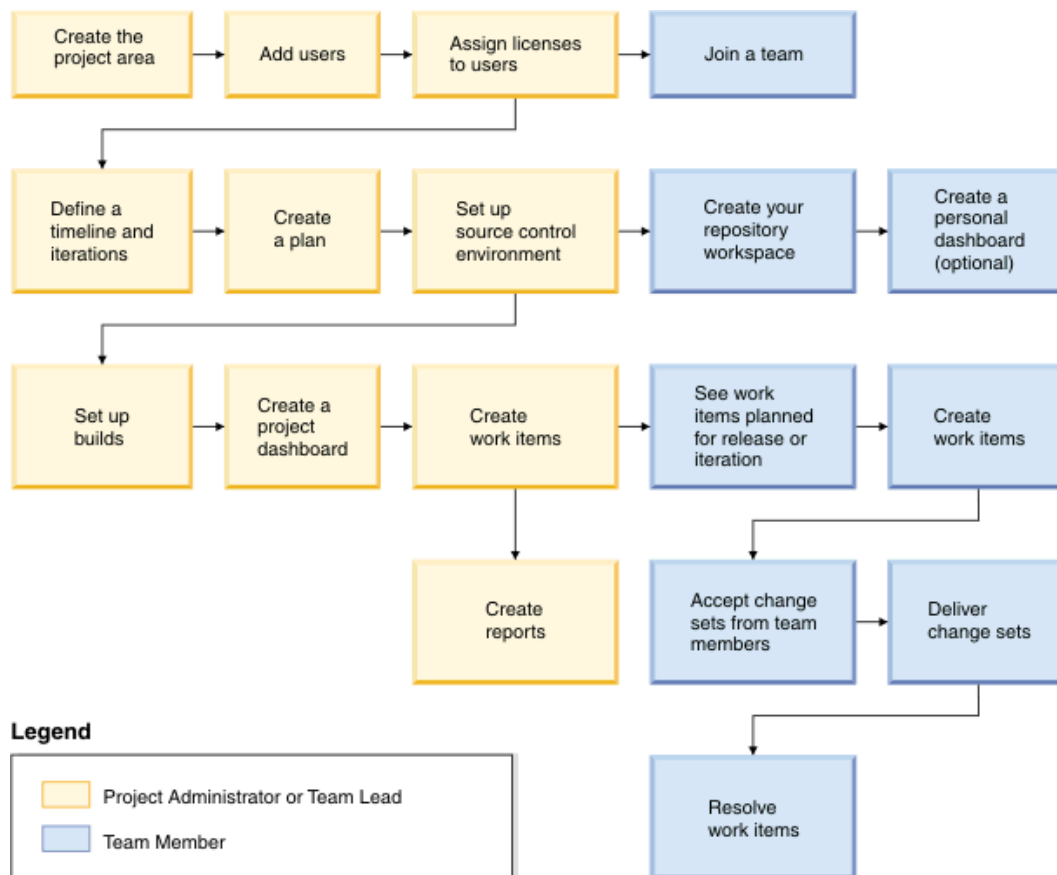


FIGURA 2.2.3: Flusso di lavoro tipico di RTC

### 2.2.3.1 Vantaggi e svantaggi

Per analizzare i vantaggi e gli svantaggi di questo strumento, viene fatto un confronto con Git.



### Modello di repository

**Git:** distribuito. I vari *repository* agiscono come peer e gli utenti hanno solitamente una copia locale con tutta la storia dei cambiamenti in aggiunta a quella in cui stanno lavorando.

**RTC:** *client-server*. Gli utenti accedono ad un *repository* principale (lo *stream*) tramite un *client*; tipicamente in locale viene conservata solamente la copia di lavoro del progetto e i cambiamenti devono essere inviati allo *stream* prima di essere propagati agli altri utenti.

Il vantaggio che un sistema centralizzato ha rispetto ad uno distribuito è che lo spazio richiesto per mantenere in locale la copia del progetto è limitato alle sole dimensioni dei file dello stesso. Questo comporta però la necessità di essere collegati in rete anche solo per consultare le modifiche apportate ad un file, che non sono disponibili in locale. Questo comportamento presenta anche il seguente svantaggio: se per un progetto di un cliente lo standard rilasciato non è aggiornato con le ultime modifiche presenti sullo *stream*, un nuovo componente del team che dovrà lavorare su quel progetto scaricherà una versione diversa da quella del cliente. Non c'è pertanto la sicurezza che quanto modificato funzioni effettivamente, se non aggiornando il pacchetto standard interamente, operazione rischiosa se fatta da tale nuova persona, che può non conoscere approfonditamente tutte le personalizzazioni del cliente.

Per ovviare a questo problema sono possibili varie strategie: la prima è un passaggio manuale del *workspace* locale del progetto tra lo sviluppatore che si è occupato del cliente in questione e il nuovo arrivato. Questo, oltre a non essere sempre possibile, presenta alcuni problemi: nel caso in cui non si voglia aggiornare tutto il pacchetto standard del cliente con la versione corrente dello *stream*, ma solo con alcune modifiche (correttive ad esempio), diventa necessario comunicare agli altri detentori del progetto quali work item sono stati scaricati o che modifiche sono state fatte, in modo tale che possano essere riportate in tutte le versioni locali. Una operazione simile è del tutto contraria alle logiche e alle regole dei sistemi di versionamento. La seconda opzione è quella di creare nello *stream* dei clienti una copia dei progetti standard, cosicché ogni progetto cliente sia indipendente. Questa strategia richiede però un impegno maggiore, soprattutto in termini di manutenibilità. Una ulteriore possibilità è quella di creare delle *release* per i vari progetti clienti, in modo tale che il nuovo componente possa scaricare lo *stream* standard alla *release* del cliente. Questa andrà revisionata man mano che vengono effettuati gli aggiornamenti. Lo svantaggio di questa modalità è che nello *stream* del progetto standard emergono tante *release* quanti sono i clienti e, per numero molto ampio, potrebbe appesantire lo *stream* stesso. Nonostante ciò, quest'ultima strategia sembra essere la più idonea per gestire questo genere di situazioni.

### Gestione della concorrenza

**Git:** *merge*. Le modifiche possono avvenire in modo concorrente e l'utente è avvisato di possibili conflitti quando vuole eseguire un aggiornamento del *repository*. Questo conflitto può essere risolto direttamente dal sistema o manualmente dall'utente per poi inviare la nuova versione agli utenti.

**RTC:** *merge* o *lock*. Sebbene sia disponibile anche il modello *lock*, in cui le modifiche non sono permesse finché l'utente non chiede e riceve il *lock* esclusivo sul file, per il B2B è utilizzato il sistema di *merge*, come in Git.

La scelta di utilizzare il modello di *merge* è sicuramente la migliore, in quanto permette di collaborare contemporaneamente sugli stessi file, il che rappresenta un notevole vantaggio nello sviluppo dei progetti.

### Metodo di salvataggio

**Git:** *snapshot*. Vengono salvati per ogni versione i file interi, compressi per ottimizzare e ridurre la dimensione complessiva dell'albero.

**RTC:** *change set*. Vengono salvati solamente i cambiamenti tra revisioni.

Sebbene con il modello di salvataggio utilizzato da Git le varie versioni dei file sono accessibili in modo immediato, il modello a *change set* richiede di fatto meno spazio sul disco. Nello specifico, nello sviluppo del B2B risulta più un vantaggio questo risparmio rispetto al tempo di accesso a revisioni precedenti.

### Integrazione con gli IDE

**Git:** ampio supporto.

**RTC:** Eclipse e Visual Studio.

L'ampio supporto che gli IDE offrono verso Git rappresenta un grosso svantaggio per RTC, il quale offre *plug-in* solamente per Eclipse e Visual Studio. Nel caso di utilizzo di altri ambienti, quindi, questo sistema di versionamento non è utilizzabile.

## 2.2.4 Tecnologie utilizzate

L'intero B2B è sviluppato sulla piattaforma [JavaEE \(Enterprise Edition\)](#). In particolare sono utilizzati Java e il *framework* Spring per il *back-end*, JSF e Primefaces per il *front-end*. Apache Tomcat è invece il contenitore [servlet open source](#) utilizzato come piattaforma di esecuzione per il portale. Esso non può essere definito un *application server* in quanto non implementa la specifica JavaEE, ma solamente il supporto verso servlet e [JSP \(JavaServer Pages\)](#).

### 2.2.4.1 Ambiente di sviluppo

L'ambiente di sviluppo utilizzato per il B2B è Eclipse. Questo strumento, sebbene sia *open source*, non è immediato come l'IDE da me utilizzato in ambiente universitario, IntelliJ. La principale differenza rilevata nell'utilizzo di questi due strumenti è la capacità di interpretare il contesto. IntelliJ è di gran lunga più "intelligente" e in grado di facilitare il programmatore nel suo lavoro. Eclipse è invece molto più statico e funzionalità come il completamento o il debug risultano complessivamente meno rapide ed intuitive. Gli unici svantaggi di IntelliJ rispetto ad Eclipse sono che il primo è a pagamento nella sua versione Ultimate, non supporta RTC e per fornire funzionalità avanzate come quelle che effettivamente offre, richiede caratteristiche hardware maggiori per l'esecuzione. Complessivamente, tra i due continuo a ritenere IntelliJ molto più avanzato.

### 2.2.4.2 Back-end

Il *back-end* del B2B è sviluppato in Java. Questo rappresenta un vantaggio soprattutto per il contesto di sviluppo, il B2B. Essendo strettamente legato ad un sistema gestionale, è istintivo pensare ad oggetti. In particolare, se combinato al gestionale Galileo offerto dall'azienda, questa caratteristica emerge maggiormente. Per quanto riguarda l'aspetto web, l'utilizzo della JavaEE permette di creare applicazioni portabili e scalabili, in grado di interfacciarsi con tecnologie obsolete ma non rimpiazzabili. Un'applicazione server JavaEE fa sì che il programmatore possa concentrarsi più sulla logica dei componenti piuttosto che sull'infrastruttura. Il back-end del B2B fa uso anche del *framework* Spring (in particolare del modulo Core) per la sua implementazione della *dependency injection*, uno degli aspetti primari dell'*Inversion of Control (IoC)*: contrariamente al comportamento standard, è una configurazione esterna che si occupa di "iniettare" le dipendenze al programma principale.



FIGURA 2.2.4: Creazione dei *bean* tramite *dependency injection*

Un vantaggio nell'utilizzo di Java per il back-end è l'ampio numero di librerie a disposizione. Oltre a quelle fornite di default dalla piattaforma JavaEE, esistono librerie in grado di fare praticamente qualunque cosa. In questo modo è possibile integrare nel portale funzionalità anche molto complesse o delicate, come la gestione dei pagamenti online o la geolocalizzazione tramite Google Maps. Un ulteriore punto a favore è la possibilità di utilizzare il *multithreading* per operazioni che potrebbero richiedere molto tempo, senza la necessità di fornire un esito al client. Esempi di utilizzo di questa caratteristica nel B2B sono lo *scheduler* per task di sincronizzazione e l'invio di email.

Lo svantaggio principale nell'utilizzo di una tecnologia come Java è che per qualsiasi modifica al codice è necessario ricompilare le classi. Questo significa che la correzione di un errore, anche solo di una lettera, in una classe richiede che tale classe sia ricompilata, quindi deve essere interrotta l'esecuzione del server, sostituito il file e riavviato il server. Questa operazione, sebbene possa richiedere pochi minuti, è comunque invasiva se si considera che il portale in quei minuti risulta off-line.

### 2.2.4.3 Front-end

Per quanto riguarda il front-end, esso è sviluppato con JSF, Primefaces e Less per lo stile.

Partendo da JSF, di seguito sono elencati alcuni dei principali vantaggi nel suo utilizzo:

- il primo vantaggio è la perfetta integrazione nella JavaEE, essendo questa tecnologia parte della specifica;
- data la sua natura, permette di creare componenti riutilizzabili, aumentando di fatto la produttività e la consistenza del progetto;
- ha un buon supporto delle espressioni [Expression Language \(EL\)](#);
- definisce i concetti di “validatore” e “convertitore”, per utilizzare oggetti complessi in componenti tipici del web, solitamente di input.

Sebbene alcuni di questi punti siano importanti per lo sviluppo di un progetto JavaEE, vi sono alcuni svantaggi altrettanto influenti:

- Non esistono report sulle performance del framework, che risulta non idoneo per applicazioni che richiedono alte prestazioni. Tramite strumenti di sviluppo come il debugger, una delle caratteristiche che emergono quando viene caricata una pagina è che un metodo richiamato in una sola linea della pagina, viene invocato almeno due, se non più volte durante il caricamento della stessa. Questo può risultare molto oneroso se il metodo in questione effettua operazioni lunghe o che hanno *side-effect* su qualche altro componente.
- Ogni pulsante o link cliccato comporta una chiamata POST. Dover eseguire il *submit* di una form anche per la navigazione è completamente scorrelato dalla logica del web e rende il codice complesso anche per operazioni semplici.
- Non è scalabile.
- Anche se a livello architetturale rispetta il *design pattern* MVC, questo non avviene a livello di pagina, dove la presentazione (la pagina XHTML) è mescolata al contenuto (le espressioni EL che accedono ai *bean*) e al comportamento (gli script JavaScript sono inseriti direttamente tra tag).
- Una conseguenza diretta del comportamento di JSF è che molte funzionalità del browser non sono utilizzabili. Ad esempio non è possibile salvare una pagina tra i preferiti, o, ancor peggio, non è possibile utilizzare il tasto *back*: una violazione, questa, in termini di usabilità ed accessibilità non di poco conto.

Direttamente correlato a JSF è Primefaces, un framework per costruire l'interfaccia utente degli applicativi JavaEE. Uno dei vantaggi principali è che questa libreria mette a disposizione moltissimi componenti, anche complessi, pronti all'uso. Il problema è che spesso questi componenti che usati da soli lavorano egregiamente, messi insieme iniziano a non funzionare correttamente. Un altro aspetto importante per questo framework è la documentazione: mentre i componenti grafici sono ampiamente documentati e presentati sul web in una pagina dedicata ([www.primefaces.org/showcase](http://www.primefaces.org/showcase)), per tutto ciò che riguarda le funzioni JavaScript non esiste nulla. Il suo utilizzo è quindi dettato dal fatto che al momento è probabilmente la libreria migliore e più completa di componenti per JSF e ciò comporta di fatto un aumento della curva di apprendimento per utilizzare in modo consapevole il front-end del B2B.

Questo tempo richiesto non favorisce lo sviluppo nel contesto di 4words, dove fin da subito è richiesta la capacità di destreggiarsi in un progetto ampio ed in continua evoluzione.

Per concludere con questa categoria di tecnologie, per lo stile grafico delle pagine è utilizzato Less, un preprocessore CSS che estende il normale linguaggio, permettendo l'utilizzo di funzioni, operatori e variabili, la nidificazione delle istruzioni, la creazione di "mixin" e numerose altre caratteristiche che rendono il codice più facile da scrivere, mantenere e comprendere.<sup>3</sup> Ad esso è affiancato Bootstrap, la libreria per il *responsive* più usata. Il primo vantaggio di uno strumento come Less è che permette di definire lo stile del B2B in modo templatizzato, ovvero per le personalizzazioni più semplici, in cui è sufficiente cambiare i colori del portale, la modifica è immediata: basta cambiare i codici dei colori definiti nelle variabili, mentre in un classico CSS sarebbe stato necessario ricercare tutte le istruzioni dove tali colori sono utilizzati. Un altro vantaggio è dato proprio dalle sue funzionalità: funzioni e istruzioni innestate diminuiscono nettamente le linee di codice che il programmatore deve scrivere per ottenere lo stile voluto. Tutto questo però comporta una serie di svantaggi comuni alla maggior parte degli strumenti di questo genere: innanzitutto il CSS va compilato e ciò richiede una attività automatizzata o manuale che se ne occupi. Il secondo problema è che il foglio di stile risultante, sebbene compresso da ulteriori strumenti, è ridondante, proprio a causa delle istruzioni innestate e tradotte. Inoltre, l'associazione con Bootstrap e Primefaces rende a volte complicato far prevalere le proprie istruzioni a quelle imposte dalle librerie, che fanno un ampio uso di istruzioni quali `!important`, costringendo dunque all'utilizzo di istruzioni altrimenti non necessarie, o ancor peggio di definizioni di stile in-line, direttamente nella pagina. Tutto ciò è aggravato dal fatto che se per qualche motivo un componente dovesse modificare direttamente il file CSS, queste modifiche sarebbero perse alla successiva ricompilazione.

L'utilizzo di queste tecnologie, quindi, è stato principalmente imposto dalla scelta di utilizzare il framework JSF, il quale, nonostante la sua perfetta (e sola) integrazione con il back-end Java, presenta alcune gravi mancanze rispetto alle convenzioni sempre più importanti del web. Il risultato è così un portale non completamente plasmato secondo la volontà dello sviluppatore, costretto in qualche modo a sottostare al funzionamento delle tecnologie utilizzate, che da strumento per realizzare idee diventano invece la limitazione alla sua creatività.

---

<sup>3</sup>Wikipedia. URL: <https://it.wikipedia.org>.



## Capitolo 3

# Il progetto stage

In questo capitolo vengono presentati i due progetti-clienti su cui si è incentrato lo stage.

### 3.1 Primo progetto

Il primo progetto graficamente rispecchia molto la versione standard del B2B, ma di fatto include varie funzionalità realizzate appositamente per il cliente. Innanzitutto, essendo questo portale destinato ad agenti e clienti selezionati, per poterlo utilizzare è necessario essere iscritti. L'iscrizione può avvenire in 2 modalità:

- l'amministratore crea un account per l'utente, il quale al primo accesso dovrà cambiare password;
- l'utente effettua una richiesta di registrazione, che l'amministratore dovrà approvare per rendere attivo l'account. Questa seconda modalità è attivabile tramite il pannello di configurazione dello stesso amministratore.

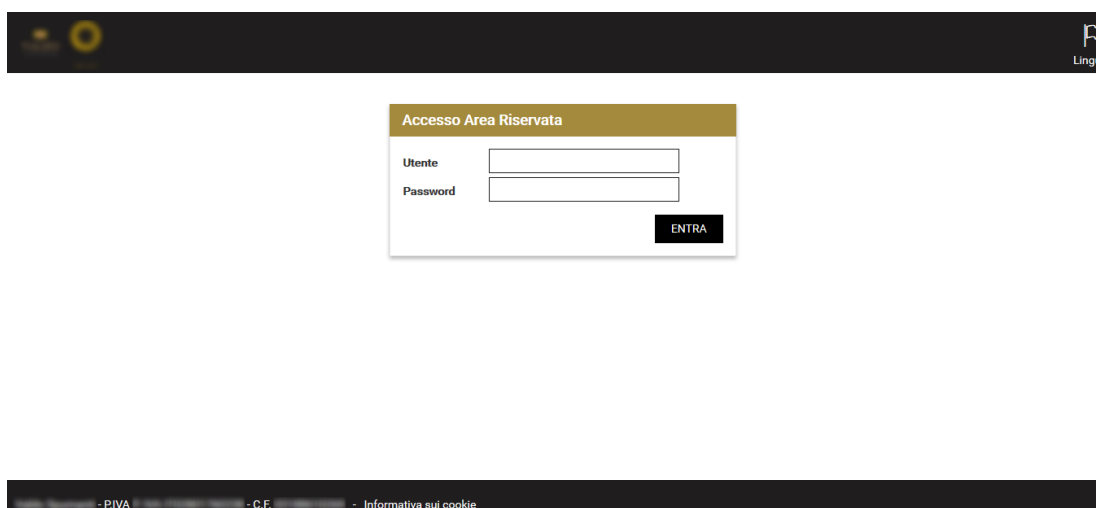


FIGURA 3.1.1: Pagina di accesso al portale

Nel corso delle settimane di stage, sono state svolte alcune implementazioni sulla base di richieste del cliente. Generalmente i lavori svolti per questo progetto erano basati su analisi svolte precedentemente al mio arrivo, in quanto parti dei preventivi presentati al cliente. Prima di iniziare a modificare il codice, mi è tuttavia stato

chiesto di presentare in forma di analisi il flusso delle operazioni necessarie per raggiungere l'obiettivo. Le richieste per questo progetto erano legate principalmente all'aggiunta o modifica di informazioni nei risultati o nei parametri di ricerca delle interrogazioni.

Vengono ora presentate alcune delle modifiche effettuate su questo progetto.

**Codice cliente** Il codice cliente è un'informazione chiave nel B2B, in quanto contribuisce ad costruire ed individuare gli ordini. Dal codice cliente dipendono il tipo dell'ordine, i listini, gli sconti e le promozioni. Per questo motivo poter effettuare ricerche tramite questo codice agevola gli agenti e l'amministrazione. Una delle prime richieste è stata proprio di introdurre questa informazione nei parametri e nei risultati delle interrogazioni. Per agevolare l'utente nell'inserimento di informazioni corrette, è stato predisposto un campo di testo "autocompletante". Una volta digitati tre caratteri nel campo di input viene eseguita una *query* di ricerca nel database per estrarre la lista dei clienti con codice o descrizione (la ragione sociale in termini del gestionale) contenenti la stringa cercata. Questa *query* deve essere il più ottimizzata possibile, in quanto l'utente non deve subire un rallentamento per un'azione come la ricerca, che ha come scopo quello di far raggiungere più rapidamente gli obiettivi. Un approccio alternativo poteva essere quello di caricare la lista di tutti gli utenti nel *backing bean*, per poi utilizzare i metodi offerti dall'interfaccia `List` per ottenere la sottolista ricercata.

## 3.2 Secondo progetto

## 3.3 I due progetti a confronto



## Capitolo 4

# Conclusioni



# Acronimi

**API** Application Programming Interface.

**B2B** Business-to-Business.

**B2C** Business-to-Customer.

**BI** Business Intelligence.

**CI** Customer Intelligence.

**CMS** Content Management System.

**CPM** Corporate Performance Management.

**DAO** Data Access Object.

**ERP** Enterprise Resource Planning.

**IBM** International Business Machines Corporation.

**IDE** Integrated Development Environment.

**iOS** iPhone Operating System.

**IVA** Imposta sul Valore Aggiunto.

**JSF** JavaServer Faces.

**JSP** JavaServer Pages.

**MVC** Model View Controller.

**PMI** Piccole e Medie Imprese.

**PPC** Pay Per Click.

**REST** REpresentational State Transfer.

**RTC** Rational Team Concert.

**SEO** Search Engine Optimization.

**SOAP** Simple Object Access Protocol.

**XHTML** eXtensible HyperText Markup Language.



# Glossario

**Android** Sistema operativo per dispositivi mobili sviluppato da Google Inc. e basato sul kernel Linux; non è però da considerarsi propriamente un sistema unix-like o una distribuzione GNU/Linux, dato che la quasi totalità delle utilità GNU è sostituita da software in Java.

**AS/400** Minicomputer sviluppato dall'IBM per usi prevalentemente aziendali, come supporto del sistema informativo gestionale. Attualmente viene utilizzato soprattutto nelle moderne reti di computer come server di applicazioni software, tipicamente di tipo gestionale o comunque di business management, o server di rete (internet/intranet).

**Beacon Award** Il programma IBM Beacon Awards premia i Business Partner IBM che offrono soluzioni straordinarie che consentono di ottenere il valore aziendale e trasformare il modo di agire di clienti e settori. I premi del 2017 riconosceranno i risultati conseguiti in una serie di aree di soluzioni – inclusi IBM Watson, Cloud e Analytics – per condurre i clienti nell'era cognitiva.

**breadcrumb** Tecnica di navigazione usata nelle interfacce utente. Il loro scopo è quello di fornire agli utenti un modo di tener traccia della loro posizione in documenti o programmi.

**Business Intelligence** Con la locuzione business intelligence (BI) ci si può solitamente riferire a un insieme di processi aziendali per raccogliere dati ed analizzare informazioni strategiche, la tecnologia utilizzata per realizzare questi processi e le informazioni ottenute come risultato di questi processi.

**business-to-business** Business-to-business, spesso indicato con l'acronimo B2B, in italiano commercio interaziendale, è una locuzione utilizzata per descrivere le transazioni commerciali elettroniche tra imprese, distinguendole da quelle che intercorrono tra le imprese e altri gruppi, come quelle tra una ditta e i consumatori/clienti individuali (B2C, dall'inglese Business to Customer o Business to Consumer, in italiano vendita al dettaglio) oppure quelle tra una impresa e il governo (B2G, dall'inglese Business to Government, lett. "azienda-verso-governo) see.

**business-to-customer** Con Business-to-customer o Business-to-consumer, spesso abbreviato in B2C, si indicano le relazioni che un'impresa commerciale detiene con i suoi clienti per le attività di vendita e/o di assistenza. Questa sigla è utilizzata soprattutto quando l'interazione tra impresa e cliente avviene tramite internet, ovvero nel caso del commercio elettronico.

**Corporate Performance Management** Con Corporate Performance Management (CPM) si intende un insieme di processi per la gestione, la misurazione e il controllo delle performance aziendali, a seguito dell'identificazione degli obiettivi

da raggiungere in un dato periodo. In tal senso è da considerarsi sinonimo di Business Performance Management (BPM) e Enterprise Performance Management (EPM).

**Customer Intelligence** Il processo di recupero e analisi di informazioni riguardanti i clienti: i loro dettagli e le loro attività, con l'obiettivo di costruire relazioni con il cliente profonde ed solide e di migliorare la strategia di marketing.

**cybersecurity** Sottoclasse del più ampio concetto di information security. Per cybersecurity si intende infatti quell'ambito dell'information security prettamente ed esclusivamente dipendente dalla tecnologia informatica. Nell'utilizzare il termine cybersecurity si vuole intendere, in particolare, un approccio mirato ad enfatizzare non tanto le misure di prevenzione (ovvero quelle misure che agiscono riducendo la probabilità di accadimento di una minaccia), ma soprattutto le misure di protezione (ovvero quelle misure che agiscono riducendo la gravità del danno realizzato da una minaccia).

**Eclipse** PrimeFaces è una suite di componenti open source per la realizzazione dell'interfaccia utente di applicazioni web basate sulla tecnologia Java Server Faces. È sviluppata dalla PrimeTek.

**enterprise resource planning** Enterprise resource planning (letteralmente "pianificazione delle risorse d'impresa", spesso abbreviato in ERP) è un sistema di gestione, chiamato in informatica sistema informativo, che integra tutti i processi di business rilevanti di un'azienda (vendite, acquisti, gestione magazzino, contabilità etc.).

**facelets** Sistema di template per il web open source, tramite costrutti XML. È la tecnologia utilizzata per creare l'interfaccia utente in JSF.

**feedback** Giudizio contenente possibili migliorie e segnalazioni di errori, inviato allo sviluppatore di un'applicazione da un utente che la collauda.

**Git** Sistema di controllo di versione per il tracciamento dei cambiamenti nei file e per il coordinamento del lavoro di più persone sugli stessi.

**Inversion of Control** Pattern per cui un componente di livello applicativo riceve il controllo da un componente appartenente a un libreria riusabile. Questo schema ribalta quello tradizionale della programmazione procedurale, dove il codice applicativo svolge i propri compiti richiamando (e quindi passando il controllo a) procedure di libreria.

**iOS** Sistema operativo sviluppato da Apple per iPhone, iPod touch e iPad.

**JavaEE** Enterprise Edition della Java Platform, anche nota come J2EE (Java 2 Enterprise Edition), è una specifica le cui implementazioni vengono principalmente sviluppate in linguaggio di programmazione Java e ampiamente utilizzata nella programmazione Web.

**Jenkins** Server di automatizzazione open source, aiuta ad automatizzare parti del processo di sviluppo del software tramite integrazione continua.

**Magento** CMS open source per l'e-commerce lanciato il 31 marzo 2008.

**Maven** Strumento per la compilazione automatizzata usato principalmente per i progetti Java.

**Open Power Foundation IBM** La Open Power Foundation è una collaborazione attorno ai prodotti Power Architecture iniziata da IBM e annunciata come l'“Open Power Consortium”. L'obiettivo è di permettere ai venditori di ambienti server di costruire server, reti e storage hardware personalizzati per futuri data centers e cloud computing.

**Pay Per Click** Il pay per click (PPC) è una modalità di acquisto e pagamento della pubblicità online; l'inserzionista paga una tariffa unitaria in proporzione ai click (click-through rate), ovvero solo quando un utente clicca effettivamente sull'annuncio pubblicitario. Un esempio di pubblicità pay per click è rappresentato dal keyword advertising, cioè annunci sponsorizzati che compaiono a lato dei risultati "puri" dei motori di ricerca.

**Power** IBM Power Systems è la linea di server basati su tecnologie open e pensati per le applicazioni mission-critical.

**Prestashop** CMS open source utilizzato per realizzare siti di e-commerce. Nasce nel 2007 e, a differenza dei CMS più “generici” diffusi all'epoca della sua prima release (WordPress e Joomla!), Prestashop è interamente pensato per lo sviluppo e la gestione dell'e-commerce..

**RPG** Linguaggio di programmazione nativo per minicomputer IBM della serie iSeries, denominata anche, più comunemente, AS/400.

**Search Engine Optimization** Con il termine ottimizzazione per i motori di ricerca (in lingua inglese Search Engine Optimization, in acronimo SEO) si intendono, nel linguaggio di internet, tutte quelle attività volte a migliorare la visibilità di un sito web sui motori di ricerca al fine di migliorare (o mantenere) il posizionamento nelle pagine di risposta alle interrogazioni degli utenti del web. A sua volta, il buon posizionamento di un sito web nelle pagine di risposta dei motori di ricerca è funzionale alla visibilità dei prodotti/servizi venduti.

**servlet** Oggetti scritti in linguaggio Java che operano all'interno di un server web.

**software house** Azienda specializzata principalmente nella produzione di software e applicazioni.

**Stakeholder** con il termine stakeholder (o portatore di interesse) si indica genericamente un soggetto (o un gruppo di soggetti) influente nei confronti di un'iniziativa economica, che sia un'azienda o un progetto. Fanno, ad esempio, parte di questo insieme: i clienti, i fornitori, i finanziatori come banche e azionisti (o shareholder), i collaboratori, dipendenti ma anche gruppi di interesse locali o gruppi di interesse esterni, come i residenti di aree limitrofe all'azienda e le istituzioni statali relative all'amministrazione locale.

**tag** Un tag (cioè etichetta, marcatore, identificatore) è una parola chiave o un termine associato a un'informazione (un'immagine, una mappa geografica, un post, un video clip, ecc.), che descrive l'oggetto rendendo possibile la classificazione e la ricerca di informazioni basata su parole chiave.

**wizard** Procedura informatica, generalmente inglobata in una applicazione più complessa, che permette all'utente di eseguire determinate operazioni (solitamente complesse) tramite una serie di passi successivi.



# Bibliografia

- [1] *E-commerce statistics for individuals*. Eurostat. 2016. URL: [http://ec.europa.eu/eurostat/statistics-explained/index.php/E-commerce\\_statistics\\_for\\_individuals](http://ec.europa.eu/eurostat/statistics-explained/index.php/E-commerce_statistics_for_individuals).
- [2] Riyadh Eid, Myfanwy Trueman e Abdel Moneim Ahmed. «A cross-industry review of B2B critical success factors». In: *Internet Research* 12.2 (2002).
- [3] *Wikipedia*. URL: <https://it.wikipedia.org>.