

Gestione grossista Il Girasole
System Design Document
Versione 1.3



Data: 28/11/2017

Progetto: Grossista Il Girasole	Versione: 1.3
Documento: System Design Document	Data: 28/11/2017

Coordinatore del progetto:

Nome	Matricola

Partecipanti:

Nome	Matricola
Silvia Castelli	0512103054
Valeria Monzillo	0512103042

Scritto da:	Silvia Castelli e Valeria Monzillo
--------------------	------------------------------------

Revision History

Data	Versione	Descrizione	Autore
24/10/2017	1.0	Scrittura punto 1	Castelli/Monzillo
25/11/2017	1.1	Scrittura punto 2 e 3	Castelli/Monzillo
27/11/2017	1.2	Scrittura punto 4 e 5	Castelli/Monzillo
28/11/2017	1.3	Revisione	

Indice

1.	INTRODUZIONE	4
1.1.	Scopo del sistema	4
1.2.	Design goals	4
1.2.1.	Criteri di Perfomance	4
1.2.2.	Criteri di Affidabilità.....	4
1.2.3.	Criteri di Costo.....	5
1.2.4.	Criteri di Manutenzione	5
1.2.5.	Criteri per l'Utente Finale	6
1.3.	Definizioni,acronimi e terminologia	6
1.4.	Referenze.....	6
1.5.	Panoramica	6
3.1.	Panoramica	7
3.2.	Decomposizione del sistema	8
3.2.1.	Interfaccia amministratore	9
3.2.2.	Interfaccia dipendente	9
3.2.3.	Interfaccia cliente	9
3.3.	Mapping hardware/software.....	10
3.4.	Gestione dati persistenti	12
3.5.	Controllo degli accessi e sicurezza	13
3.6.	Controllo globale del sistema.....	13
3.7.	Condizioni limite.....	13
4.1.	Amministratore	14
4.1.1.	Gestione dipendenti.....	14
4.1.2.	Gestione stato ordini	14
4.2.	Dipendenti	15
4.2.1.	Gestione prodotti.....	15
4.3.	Cliente	15
4.3.1.	Gestione carrello	15
4.3.2.	Gestione acquisti	15

1. INTRODUZIONE

1.1.Scopo del sistema

Con il grande sviluppo tecnologico, rendere il proprio negozio accessibile anche online è diventato sempre più importante.

L'idea è quella di facilitare ai clienti, l'acquisto dei vari prodotti, e ai dipendenti, l'organizzazione dell'ampio magazzino.

Figura cardine dell'azienda è l'amministratore, che tramite il sistema, avrà libero accesso per gestire in maniera adeguata ed efficiente il gran numero di dipendenti e degli ordini ricevuti. Allo stesso modo, verrà affidato ai dipendenti il compito di organizzare, inserire o eliminare i prodotti in base alla loro disponibilità.

Dunque, lo scopo principale è quello di velocizzare e semplificare l'intera organizzazione dell'azienda, per aumentarne la produzione e il relativo profitto annuo.

1.2.Design goals

1.2.1. Criteri di Performance

Tempo di risposta	Obbiettivo principe è che il sistema risponda alle centinaia di richieste di carattere differente, con estrema rapidità, all'incirca una decina di secondi. Il tempo di risposta sarà, però, fortemente influenzato dalla qualità della rete utilizzata.
Troughput	Ci si propone di garantire durante, i picchi di carico di varia entità, la fluidità del sistema e il corretto funzionamento di tutte le operazione che esso dovrà effettuare, in base alle richieste dei vari utenti.
Memoria	Stabilire con precisione la memoria necessaria per l'utilizzo del sistema non è ancora possibile. Dovrà sicuramente essere installato dall'amministratore e circa una decina di utenti.

1.2.2. Criteri di Affidabilità

Robustezza	Le varie componenti del sistema devono essere in grado di garantire la sicurezza di tutti i dati in esso contenuti, soprattutto nel caso in cui un utente non autorizzato tenti di accedere al sistema o in caso di guasti.
Disponibilità	Una volta che il sistema è stato realizzato sarà disponibile ogni qualvolta che il cliente,il dipendente o l'amministratore ne richiederà l'utilizzo(eccetto nel caso di guasti temporanei).

Tolleranza all'errore	Il sistema riuscirà a garantire, in base all'entità del danno, le proprie funzionalità o almeno in parte. Verranno effettuati backup periodici, in modo da conservare i dati sensibili, nel caso in cui si verificassero guasti gravi.
Sicurezza	La sicurezza viene basata essenzialmente nei limiti di una login e una password. Il sistema, tramite funzionalità specifiche, sarà in grado di riconoscere l'utente che richiede l'accesso. In questo modo verrà garantito ai singoli (quali amministratore, dipendenti e clienti) di svolgere liberamente i compiti assegnati e gli acquisti desiderati, mostrando loro esclusivamente l'area a cui hanno libero accesso.

1.2.3. Criteri di Costo

Sviluppo	Lo sviluppo del sistema non richiede costi eccessivi riguarderanno l'affitto del server e il compenso per l'implementazione del sistema stesso.
----------	---

1.2.4. Criteri di Manutenzione

Estensibilità	E' consentito, in quanto è possibile aggiungere in futuro, nuove funzionalità al sistema (ex:come estendere il campo d'azione al di fuori del territorio nazionale) oppure creare nuove classi, con l'estensione di quelle già esistenti.
Modificabilità	Il codice sarà strutturato in maniera tale per poter essere cambiato o ampliato, nel caso in cui si presentino nuove esigenze per l'azienda.
Leggibilità	Mediante la lettura del codice il sistema è comprensibile. Infatti semplici istruzioni consentono di comprendere le funzionalità del sistema e con l'ausilio di commenti e documentazione il lavoro di comprensione è ulteriormente semplificato.
Tracciabilità dei requisiti	Tramite la tracciabilità dei requisiti sarà possibile apportare modifiche al codice e al sistema.

1.2.5. Criteri per l'Utente Finale

Usabilità	Il sistema deve essere facile da utilizzare, anche per utenti poco esperti, ma allo stesso tempo robusto e affidabile, in modo da garantire la riservatezza di tutti i dati sensibili e non, per i clienti, i dipendenti e l'amministratore.
-----------	--

1.3. Definizioni, acronimi e terminologia

-Definizioni:

Browser: Microsoft Edge, Firefox, Opera, Google Chrome

WebServer: server dove vengono memorizzate le risorse.

-Acronimi:

RAD: Requirements Analysis Document

SDD: System Design Document

DB: DataBase

DBMS: DataBase Management System

1.4. Riferenze

L'insieme del materiale di riferimento utilizzato per la realizzazione del progetto e per la stesura di questo stesso documento comprende:

- Libro di testo: B.Bruegge, A.H. Dutoit, Object Oriented Software Engineering - Using UML, Patterns and Java, Prentice Hall, 3rd edition, 2009
- Slide del docente, reperibili sulla piattaforma
- Documento RAD del sistema il Girasole

1.5. Panoramica

Questa sezione descrive l'Architettura Software proposta per il sistema in base alle seguenti voci:

- **Decomposizione in sottosistemi:** Viene effettuata una descrizione ad alto livello della suddivisione del sistema in vari sottosistemi.
- **Mapping hardware/software:** Vengono mostrare le scelte intraprese circa le piattaforme hardware e software su cui il sistema dovrà girare. Successivamente sarà necessario:
 1. Specificare le funzionalità delle varie componenti;
 2. Mappare le componenti individuate sulle piattaforme scelte.

A tal fine, la descrizione di tale voce viene affiancata ad un component diagram e ad un deployment diagram per meglio delucidare quanto suddetto.

- **Gestione dei dati Persistenti:** Consente di identificare gli oggetti persistenti.
- **Controllo degli accessi e sicurezza:** Viene specificato con quale meccanismo ci si autentica nel sistema e come esso si interfaccia con il sistema in termini di matrici di accesso.
- **Controllo Software:** Descrive come il controllo globale viene implementato, e come viene gestita la concorrenza.
- **Boundary Conditions:** Vengono descritte le condizioni limite per il sistema il Girasole quali, configurazione, accensione e spegnimento del sistema
- **Servizi dei Sottosistemi:** Vengono descritti in termini di operazioni i servizi forniti da ogni sottosistema.

2. Architettura del sistema corrente

Il sistema che verrà a crearsi, non si basa o integra nessun sistema preesistente.

3. Architettura del sistema proposto

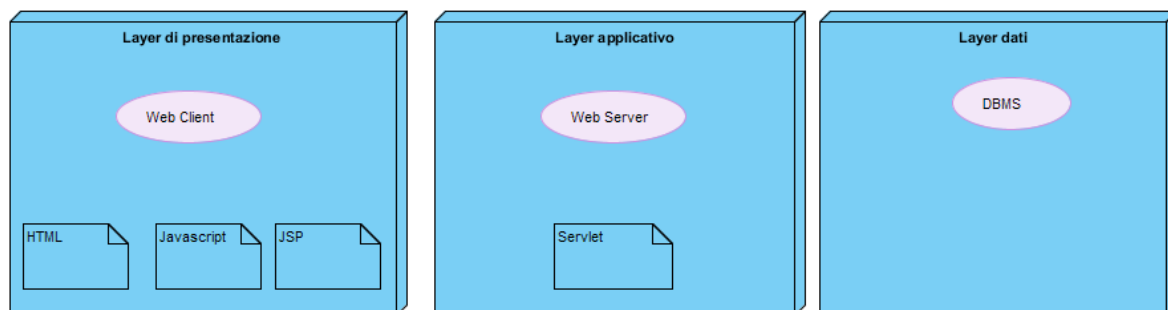
3.1. Panoramica

Andremo a creare un sistema software con un'architettura three-tier, ovvero un sistema web-based a tre livelli.

Tale architettura sarà quindi strutturata in tre strati chiamati layer:

- Layer di presentazione, implementato dal client;
- Layer applicativo, implementato dal server;
- Layer dati, composto da un DBMS per la gestione dei dati;

Questi strati potranno essere installati o sulla stessa macchina o in macchine diverse.

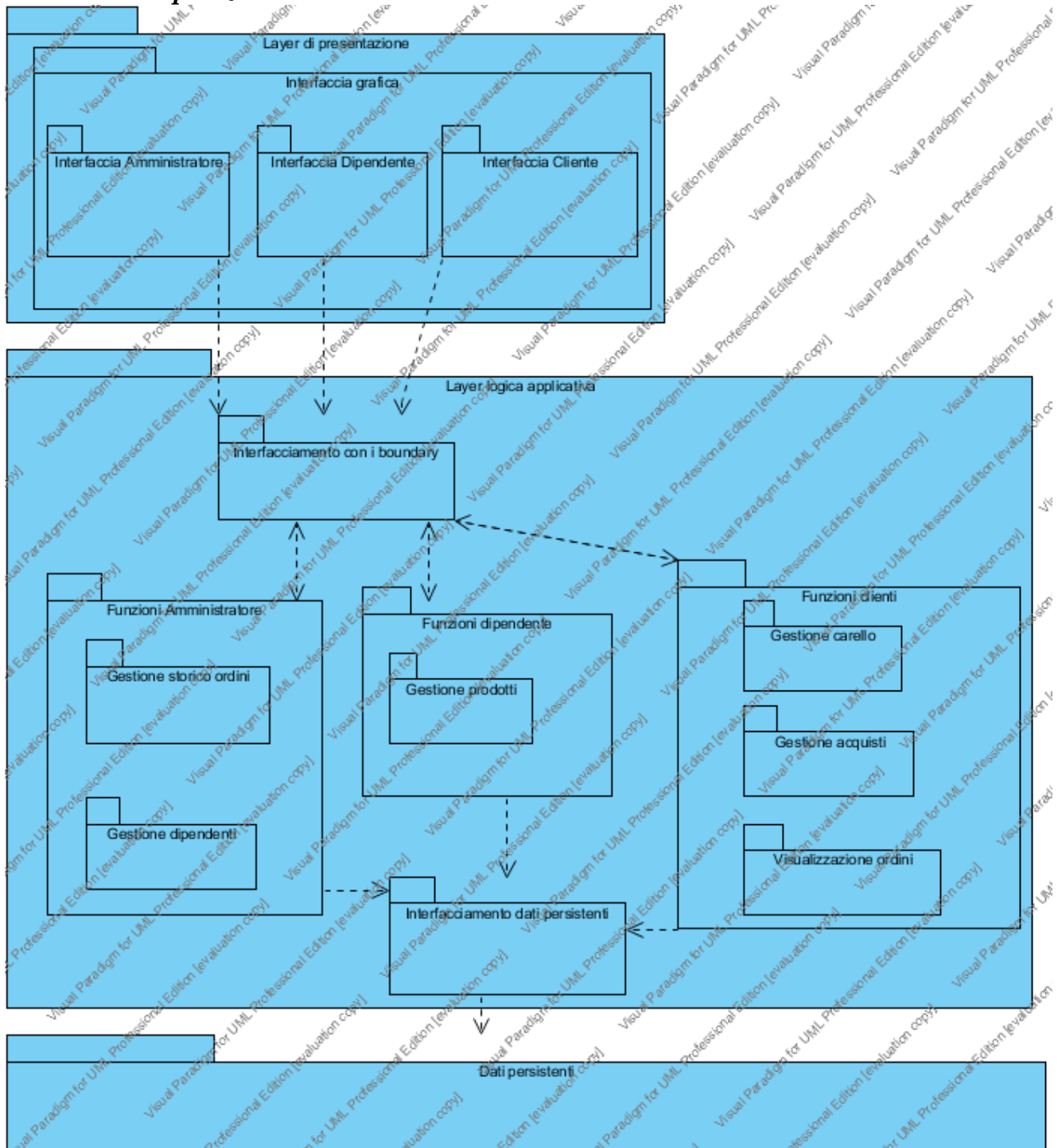


Il sistema software creato potrà essere accessibile da diversi nodi contemporaneamente in quanto sistema web-based. Infatti potrà accedervi un grosso numero di utenti garantendo persistenza dei dati e fornendo interfacce grafiche di facile intuizione.

Attraverso tale interfaccia i singoli utenti potranno interagire con il sistema in base al loro ruolo e alle funzioni che devono svolgere. Infatti avremo un'interfaccia amministratore tramite la

quale, quest'ultimo, gestirà gli ordini e i dipendenti, un interfaccia dipendente in cui verrà gestito il catalogo prodotti e infine un interfaccia cliente in cui si potrà visualizzare il catalogo prodotti e eventualmente acquistare quelli di interesse.

3.2. Decomposizione del sistema



Come già detto in precedenza il sistema sarà un software con architettura three-tier. Quindi avremo un architettura divisa in tre layer che comunicheranno tra loro seguendo le linee del client-server (il layer di presentazione è cliente del layer applicativo a sua volta cliente del layer dati). Con questa

architettura i layer sono indipendenti l'uno dall'altro e potranno essere sostituiti o modificati senza provocare cambiamenti tra loro.

I tre strati sono così divisi:

- Layer di presentazione, in cui viene rappresentata l'interfaccia utente divisa tra interfaccia amministratore, interfaccia dipendente e interfaccia cliente.
- Layer applicativo fornisce i servizi in base all'interfaccia in cui ci troviamo e in questo livello avviene anche l'interrogazione del database tramite il layer dati permettendo la generazione di contenuti dinamici e la persistenza dei dati.
- Layer dati che permette la lettura, la memorizzazione e la persistenza dei dati del sistema.

3.2.1. Interfaccia amministratore

L'interfaccia amministratore permette di far autenticare l'utente e portarlo nella pagina personale che gli permetterà di effettuare le proprie mansioni quali la gestione degli ordini (in modo da prepararli fisicamente e spedirli) e la gestione dei dipendenti in modo da tenere sotto controllo chi può apportare le modifiche al catalogo prodotti.

Funzioni amministratore

Gestione storico ordini: Permette di controllare lo stato degli ordini (in preparazione, spedito, consegnato) e cambiarlo in base allo stato reale degli ordini.

Gestione dipendenti: Permette di visualizzare la lista completa dei dipendenti con i loro dati generali e di aggiungerli e rimuoverli in caso di assunzione o licenziamento.

3.2.2. Interfaccia dipendente

Tale interfaccia dà la possibilità al singolo dipendente dell'azienda di autenticarsi e entrare nella pagina personale dove potranno aggiungere e eliminare i prodotti all'interno del catalogo.

Funzioni dipendente

Gestione prodotti: Permette di gestire il catalogo dei prodotti, aggiungendoli o rimuovendoli.

3.2.3. Interfaccia cliente

L'interfaccia cliente permette l'interazione del cliente con il sistema. Attraverso questa interfaccia l'utente senza doversi autenticare può visualizzare informazioni generali sull'azienda e visionare il catalogo dei prodotti. Se l'utente è intenzionato ad effettuare l'acquisto deve o registrarsi al sito o se già in possesso delle credenziali autenticarsi tramite il login.

Funzioni cliente

Gestione carrello: Permette di aggiungere i prodotti nel carrello, indicando la quantità, o

eventualmente cancellarli.

Gestione acquisti: Permette di acquistare i prodotti presenti all'interno del carrello.

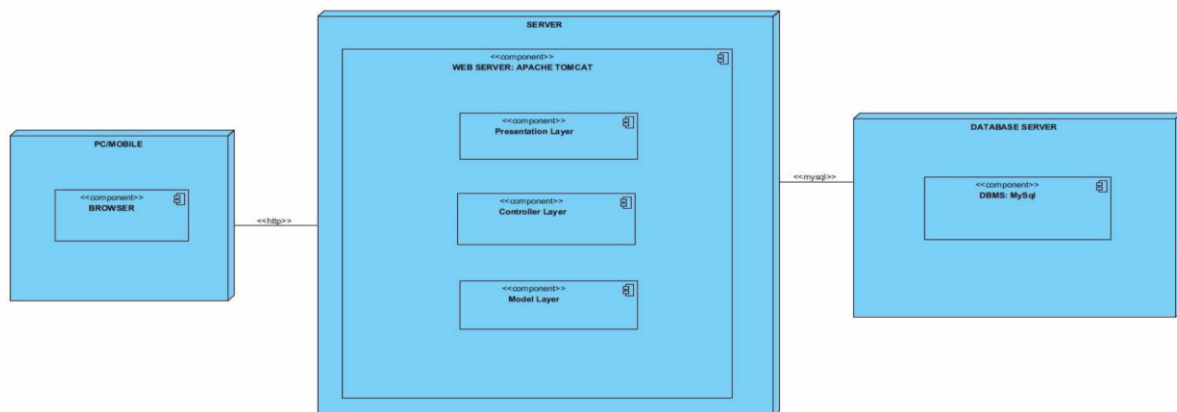
Visualizzazione ordini: Permette di visionare gli ordini già effettuati e eventualmente controllare il loro stato.

3.3.Mapping hardware/software

L'hardware proposto è costituito da un sever e dai client.

Al server si collegheranno i client (utenti che accedono al sito dell'azienda Il girasole direttamente dal browser) e il database.

Sarà il client che effettuerà richieste al server per poter eseguire le specifiche funzioni. Il tipo di utente che accederà al sito verrà determinato al momento dell'autenticazione.



Web Server

Viene utilizzato Apache Tomcat come server.

Presentation Layer

Il sistema viene usufruito dall'utente attraverso un browser già presente all'interno del dispositivo utilizzato.

Controller Layer

Le funzionalità del sistema creato sono implementate in linguaggio JAVA. Quest'ultimo tramite le funzionalità prodotte dalle Servlet verrà poi elaborato e tradotto in JSP, in tal modo i file ottenuti possono essere inviate al client.

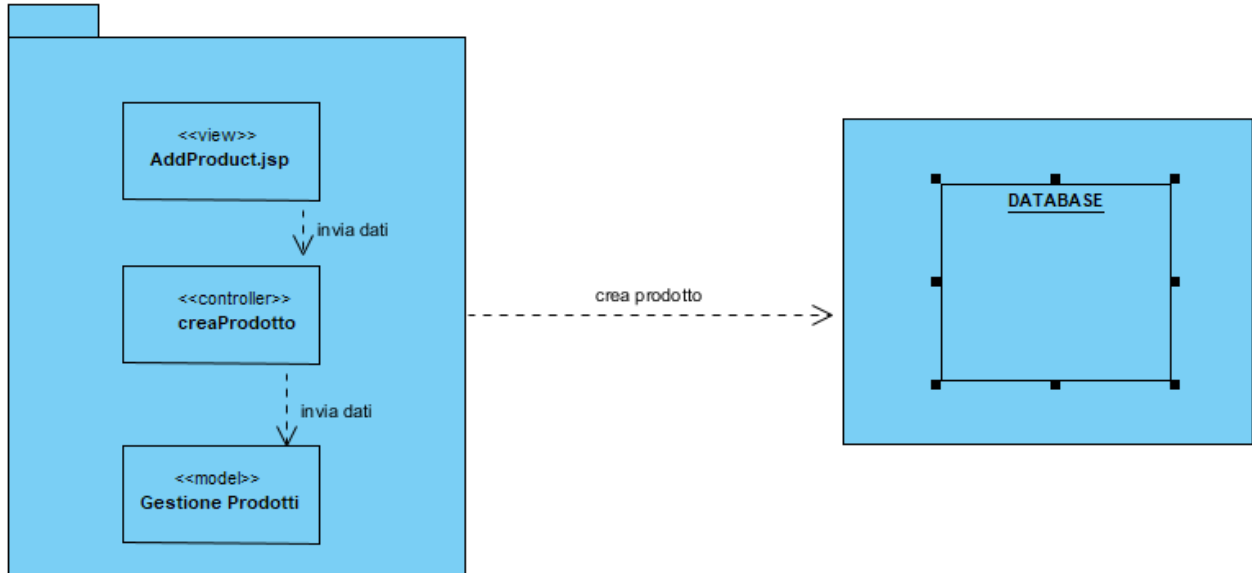
Model Layer

Layer che si occupa del collegamento con il server da parte del sistema e delle richieste di accesso e modifiche sui dati persistenti nel database.

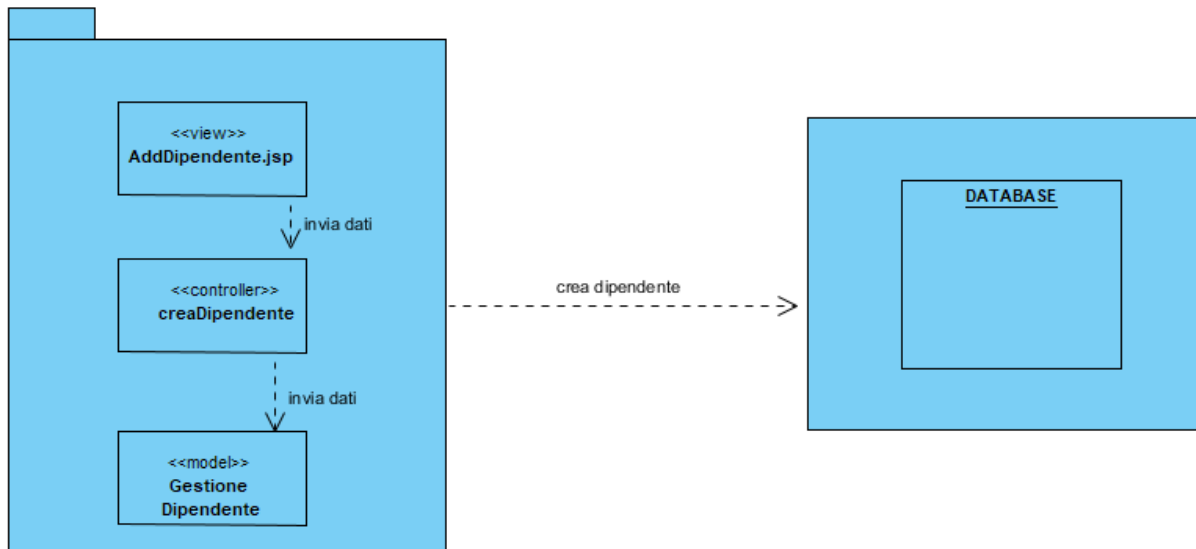
Database Server

Il DBMS usato è MySQL il quale presenta molte API che permettono l'interazione tra sistema e database.

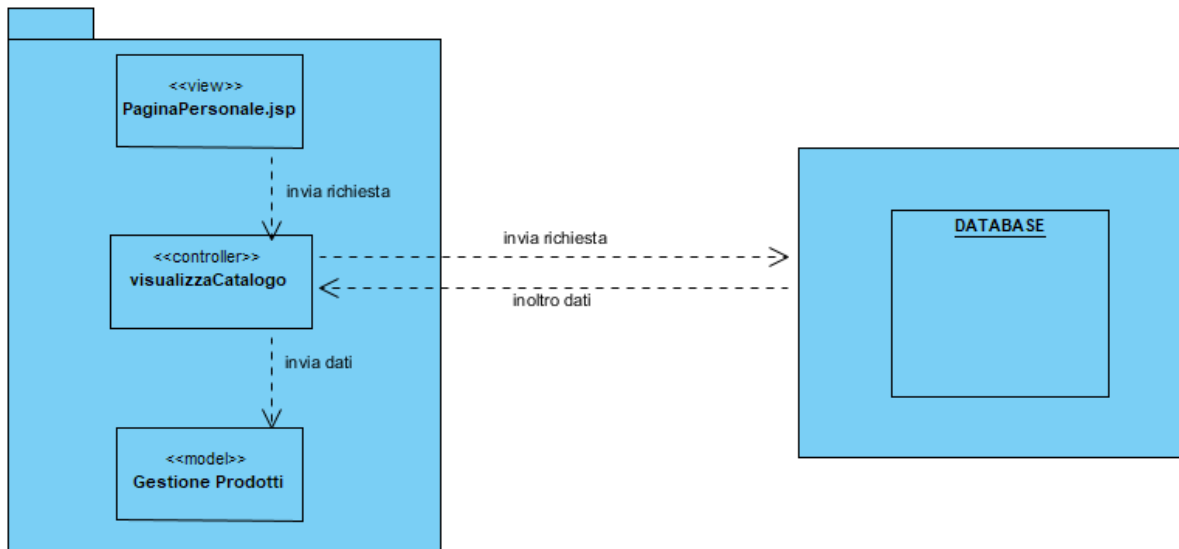
Aggiungi prodotto:Dipendente



Aggiungi dipendente: Amministratore



Visualizzazione catalogo prodotti: Dipendente/Cliente



3.4. Gestione dati persistenti

Sulla base del Requirements Analysis Document, sono stati identificati i seguenti dati persistenti.

Amministratore: Persona a capo dell'azienda che gestisce le assunzioni dei dipendenti, il loro licenziamento e quindi il loro inserimento e eventuale cancellazione dal sistema, lo storico degli ordini dove cambierà lo stato degli ordini e acquisirà le informazioni necessarie per prepararli nel reale.

Cliente: Persona che accederà al sito con lo scopo di acquistare i prodotti e a cui sarà consentito visionare il catalogo prodotti e aggiungere quelli di interesse nel carrello. Il cliente ha anche la possibilità di modificare il proprio carrello e effettuare l'acquisto.

Dipendente: Persona che ha il compito di visionare il catalogo prodotti e aggiungerli o eliminarli da quest'ultimo.

Prodotto: Merce venduta e gestita dall'azienda.

Catalogo prodotti: Lista dettagliata dei prodotti presenti nell'azienda.

Storico ordini: Lista dettagliata degli ordini in cui è possibile cambiare il loro stato (In elaborazione, spedito e consegnato).

Carrello: Contenitore che permette al cliente di visionare i prodotti che vuole acquistare.

Gestione prodotti: Gestisce il catalogo dei prodotti aggiungendoli o cancellandoli.

Gestione stato ordini: Gestisce lo stato degli ordini che possono essere in elaborazione, spediti o consegnati.

Gestione dipendenti: Gestisce l'inserimento e la cancellazione dei dipendenti nel database.

Gestione carrello: Gestisce i prodotti all'interno del carrello.

Gestione acquisti: Gestisce l'acquisto dei prodotti nel carrello.

3.5. Controllo degli accessi e sicurezza

Le funzioni del sistema sono state divise in base alla tipologia di utente riconosciuta al momento del login: Amministratore, Dipendente e Cliente.

Funzionalità dell'amministratore:

- Gestione dipendenti
- Gestione stato ordini
- Visualizzazione elenco dipendenti
- Visualizzazione storico ordini

Funzionalità del dipendente:

- Gestione prodotti
- Visualizzazione catalogo prodotti

Funzionalità del cliente:

- Visualizzazione catalogo prodotti
- Visualizzazione carrello
- Visualizzazione ordini effettuati
- Gestione carrello
- Gestione acquisti

3.6. Controllo globale del sistema

Il flusso di controllo sarà gestito nella maniera più consona ad un'applicazione web, quindi sarà di tipo event-driven. La gestione della concorrenza inoltre è un problema corposo da gestire in un approccio event-driven a causa dell'imprevedibilità degli eventi provenienti dai numerosi agenti con cui il sistema si troverà a dialogare: l'integrità dei dati è tuttavia garantita dal DBMS MySQL, tramite la semantica delle transazioni.

3.7. Condizioni limite

Inizializzazione

Il sistema lato server parte nel momento in cui è lanciato il web presentando un'interfaccia ai client, in base alle funzionalità che devono svolgere, ogni volta che vi accedono da un generico browser.

Terminazione

Il sistema lato server non deve mai terminare a meno di guasti.

Il sistema lato client termina al momento della chiusura del browser.

Fallimento

Il sistema lato server può fallire solo a causa di condizioni eccezionali quali mancanza di elettricità o guasti all'hardware (hard disk danneggiato ecc., che danneggiano permanentemente il sistema) o in caso di crash di sistema (attacchi al server dall'esterno). Il sistema lato client può fallire a causa di guasti temporanei (hardware o software), o per la caduta della linea telefonica, ma ciò non influisce in alcun modo sul lato server (che resterà comunque stabile), neanche se ci fosse stata una connessione attiva al DB, visto che il DBMS gestisce le transazioni. Per recuperare da un fallimento basta riaccendere il web server, ma ovviamente, nel frattempo, i client non possono usufruire nel sistema visto che il server è down. Essendo distribuito, potrebbe verificarsi un congestionamento della rete che provocherebbe un notevole rallentamento delle interazioni.

4. Servizi dei sottosistemi

4.1. Amministratore

4.1.1. Gestione dipendenti

Sottosistema	Gestione dipendenti
Descrizione	Sottosistema che gestisce i dipendenti all'interno del sistema
Servizi offerti	
Visualizza elenco dipendenti	Tale servizio permette la visualizzazione di tutti i dipendenti presenti all'interno dell'azienda e i dati ad essi relativi.
Inserimento nuovo dipendente	Questa funzionalità consente all'amministratore di registrare un nuovo dipendente all'interno del sistema, assegnandoli così una login e una password.
Rimozione dipendente	Tale funzionalità rimuove un dipendente dal sistema, privandolo così delle credenziali assegnategli precedentemente.

4.1.2. Gestione stato ordini

Sottosistema	Gestione stato ordini
Descrizione	Sottosistema che gestisce gli ordini ricevuti dai vari clienti.
Servizi offerti	
Visualizza elenco ordini	Tale servizio permette la visualizzazione di tutti gli ordini ricevuti e il loro relativo.
Modifica lo stato dell'ordine	Questa funzionalità consente all'amministratore di modificare lo stato dell'ordine in base alle disponibilità e le direttive dell'azienda.

4.2.Dipendenti

4.2.1. Gestione prodotti

Sottosistema	Gestione prodotti
Descrizione	Sottosistema che gestisce il catalogo dei prodotti in base alla loro disponibilità.
Servizi offerti	
Visualizza elenco prodotti	Tale servizio permette la visualizzazione di tutti i prodotti disponibili all'interno dell'azienda.
Inserimento nuovo prodotto	Questa funzionalità consente al dipendente di inserire un nuovo prodotto, compilando tutti i campi ad esso relativi.
Rimozione prodotto	Questa funzionalità permette al dipendente di rimuovere definitivamente un prodotto dal catalogo.

4.3.Cliente

4.3.1. Gestione carrello

Sottosistema	Gestione del carrello
Descrizione	Sottosistema che gestisce il carrello e il suo contenuto..
Servizi offerti	
Visualizza prodotti nel carrello	Tale servizio permette la visualizzazione di tutti i prodotti disponibili all'interno del carrello.
Rimuovere un prodotto dal carrello	Questa funzionalità permette all'utente di eliminare, se lo si desidera, uno dei prodotti precedentemente inseriti nel carrello.

4.3.2. Gestione acquisti

Sottosistema	Gestione acquisti
Descrizione	Sottosistema che gestisce gli acquisti che vengono effettuati dai vari clienti.
Servizi offerti	
Visualizza stato ordini effettuati	Tale servizio permette la visualizzazione di tutti gli ordini che il cliente ha effettuato e il corrispettivo stato.

5. Glossario

Atomicità(operazione) : consiste in un'operazione di esecuzione indivisibile dal punto di vista logico.

Backup: si indica la replicazione, su un qualunque supporto di memorizzazione, di materiale informativo archiviato nella memoria di massa dei computer.

Controller: riceve i comandi dell'utente (in genere attraverso il view) e li attua modificando lo stato degli altri due componenti.

Coerenza(dati): processo di mantenere l'uniformità dei dati.

Event-driven: programma che gestisce le risposte alle azioni generate dall'utente o dal sistema.

HTML: L'HyperText Markup Language, in informatica è il linguaggio di markup solitamente usato per la formattazione e impaginazione di documenti ipertestuali disponibili nel World Wide Web sotto forma di pagine web.

MySQL: è un Relational Database Management System (RDBMS) composto da un client a riga di comando e un server.

Model: fornisce i metodi per accedere ai dati utili all'applicazione.

JSP: In informatica JavaServer Pages, di solito indicato con la sigla JSP, è una tecnologia di programmazione Web in Java per lo sviluppo della logica di presentazione di applicazioni Web, fornendo contenuti dinamici in formato HTML.

Javascript: è un linguaggio di scripting orientato agli oggetti e agli eventi, comunemente utilizzato nella programmazione Web lato client.

Java: è un linguaggio di programmazione orientato agli oggetti a tipizzazione statica specificatamente progettato per essere il più possibile indipendente dalla piattaforma di esecuzione.

Servlet: sono oggetti scritti in linguaggio Java che operano all'interno di un server web.

View: visualizza i dati contenuti nel model e si occupa dell'interazione con utenti e agent.

Web-Server: è un'applicazione software che, in esecuzione su un server è in grado di gestire le richieste di trasferimento di pagine web di un client, tipicamente un web browser.