

Nome Progetto: Gametop
Titolo Documento: System design document



Sommario

MANAGER:	3
REVISION HISTORY:	3
SOMMARIO	2
1 OBIETTIVI DESIGN	4
1.2 Compromessi	5
1.3 DEFINIZIONI, ACRONIMI ED ABBREVIAZIONI	5
1.4 RIFERIMENTI	5
1.5 PANORAMICA	5
2.2Decomposizione del sistema	7
2.2.1 Gestione Sottosistema	7
2.3 HARDWARE/SOFTWARE MAPPING	8
2.4 GESTIONE DEI DATI PERSISTENTI	9
2.5 CONTROLLO DEGLI ACCESSI E SICUREZZA	10
3 Subsystem service	13

SYSTEM DESIGN DOCUMENT

Manager:

Professori
Prof. De Lucia Andrea

Partecipanti:

Nome	Matricola
Vincenzo Tarantino	0512108025
Luigi Sodano	0512107590

Revision History:

Data	Versione	Descrizione	Autore
18/10/2023	1.0	Prima stesura del document problem statement	Team members
30/10/2023	2.0	Stesura del document RAD	Team members
15/12/2023	3.0	Revisione del document RAD	Team members
03/01/2024	3.1	Ultima revisione del document	Team members
21/11/2023	3.2	Stesura document System Design	Team members

04/01/2024	3.3	Revisione document System Design	Team members
------------	-----	----------------------------------	--------------

1 OBIETTIVI DESIGN

Nella seguente tabella vengono illustrati gli obiettivi di design per il sistema e le relative priorità (a numeri più bassi corrispondono priorità più elevate). Per ogni obiettivo viene riportata anche l'origine facendo riferimento, in particolare, all'identificativo del requisito non funzionale ad esso associato.

Priorità	ID	Descrizione	Categoria	Origine
1	DG_1	Robustezza: Il sistema deve sopravvivere agli input errati degli utenti. In un caso del genere il sistema non deve accettare l'input e notificare l'utente invitandolo a correggere l'errore	Affidabilità	RNF_2 (Affidabilità)
3	DG_2	Sicurezza: Il sistema deve comunicare tramite protocollo HTTPS in modo da assicurare una maggiore sicurezza.	Sicurezza	RNF_2
2	DG_3	Usabilità: Il sistema deve essere facile da utilizzare senza necessariamente consultare la documentazione anche tramite l'utilizzo di un menu contestuale che permette di muoversi all'interno del sito in qualsiasi momento. I contenuti dovranno essere fruibili attraverso dispositivi sia desktop che mobile ed accessibili attraverso un numero ridotto di interazioni.	Usabilità	RNF_1 (Usabilità) RNF_2 (Affidabilità) RNF_3 (Performance) RNF_4 (Manutenibilità)
2	DG_4	Costi di sviluppo: Bisogna ridurre al minimo i costi di sviluppo del sistema	Costo	RNF_5 (Implementazione)
1	DG_5	Tempi di sviluppo: Bisogna sviluppare il sistema in tempi ridotti	Costo	RNF_5 (Implementazione)
3	DG_6	Prestazioni: Il sistema deve essere in grado di supportare fino a 1000 utenti simultaneamente e conservando le loro informazioni in una struttura dati.	Prestazioni	RNF_3 RNF_4
3	DG_7	Tempi di risposta: Il sistema deve elaborare le richieste e produrre output in meno di 2 secondi.	Prestazioni	RNF_3

2	DG_8	Leggibilità: Il codice prodotto dev'essere semplice da comprendere. Ogni metodo e campo non banale dev'essere documentato opportunamente al fine di aumentarne la comprensione	Sopportabilità	RNF_5 (Implementazione)
---	------	---	----------------	----------------------------

1.2 Compromessi

Tempo di rilascio vs Funzionalità

Il team si impegna nel consegnare il sistema completo di tutte le sue funzionalità nei tempi stabiliti.

Prestazioni vs Costi

Il team cercherà di ottenere le migliori prestazioni per non sforare il budget a disposizione.

1.3 Definizioni, acronimi ed abbreviazioni

- Piattaforma = Applicazione web
- CRUD = Create, Read, Update, Delete

1.4 Riferimenti

- Requisiti funzionali: Sezione 3.1 del Problem statement
- Requisiti non funzionali: Sezione 3.2 del Problem statement

Panoramica

Nel documento verrà effettuata l'analisi delle architetture di sistemi simili e la scomposizione in sottosistemi del sistema proposto. Verranno quindi definiti i servizi esposti da ciascun sottosistema.

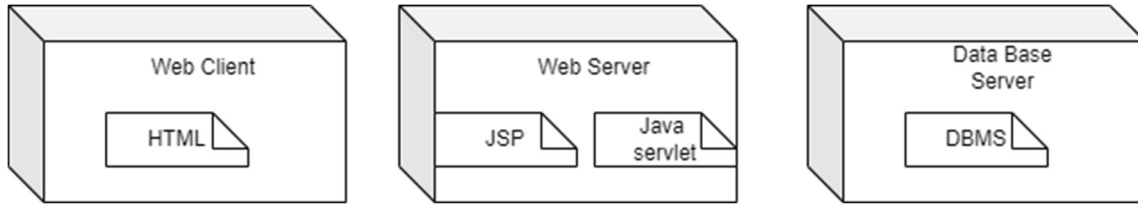
Per lo sviluppo del sistema utilizzeremo un'architettura di tipo MVC. Il pattern Model-View-Controller prevede che il software venga diviso in tre parti ognuna delle quali ha un compito diverso.

- **Il model** si occupa della gestione dei dati e quindi sarà responsabile dell'interazione con il database sottostante
- **La view** si occupa di curare l'interazione con l'utente e quindi avrà il compito di gestire la formattazione dei dati che verranno visualizzati.
- **Il controller** si occupa di elaborare i dati inviati dall'utente, passarli al model se necessario e inviare la risposta al view appropriato.

L'utilizzo del modello MVC per lo sviluppo del sistema permette di avere vari vantaggi, tra i quali la facilità nella suddivisione dei compiti tra i vari membri del team oltre che a semplificare eventuali attività di modifica che devono essere effettuate sul sistema.

All'interno del nostro sistema:

- Il model verrà realizzato utilizzando classi Java appropriate.
- La parte di view verrà implementata utilizzando pagine HTML e JavaScript.
- I control saranno realizzati tramite Servlet.



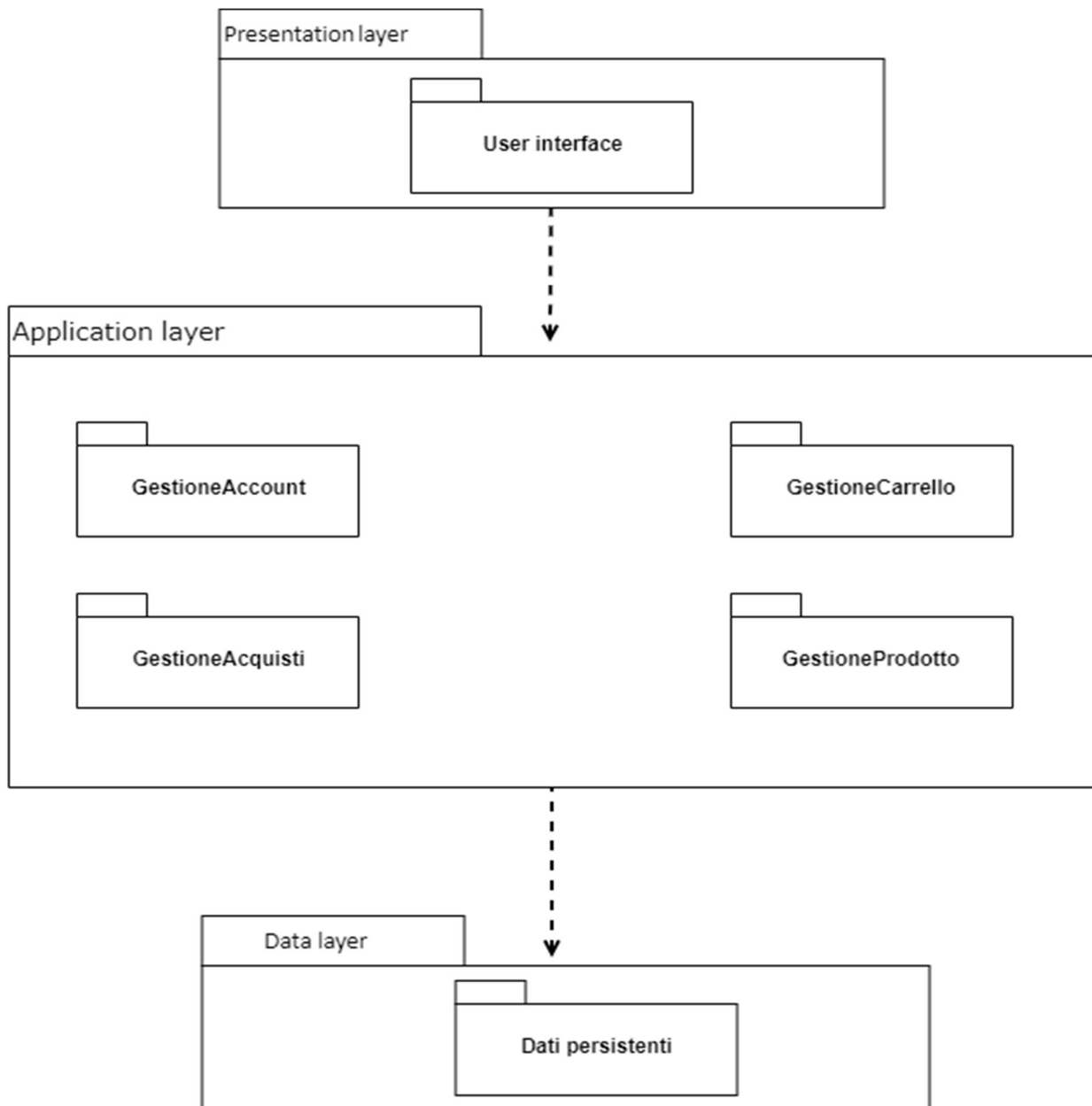
2.2 Decomposizione del sistema

Per realizzare il sistema è stata usata un'architettura three-tier.

Questo è un caso particolare di un'architettura multi-tier in cui la logica dell'applicazione viene suddivisa in tre layer:

- Presentation layer: consiste nelle interfacce grafiche e in particolare dai boundary object come i form che vengono compilati dall'utente. (Frontend, ovvero l'interfaccia che gli utenti utilizzeranno per interagire col sistema).
- Application layer: composto dagli oggetti che si occupano della gestione del controllo, dell'elaborazione dati e di notificare i cambiamenti al Presentation layer. Questo strato interagisce con il database sottostante tramite il data layer.
- Data layer: si occupa della memorizzazione dei dati persistenti e del loro recupero dal database ad esempio eseguendo delle query.

2.2.1 Gestione Sottosistema



- **User Interface:**
 1. Gestisce le interazioni tra utente e sistema attraverso le pagine JSP e servlet per i vari servizi offerti dal sistema.
- **GestioneAccount:**
 1. Il cliente può effettuare l'accesso al sistema utilizzando le proprie credenziali e può effettuare il logout. Può inoltre visualizzare lo storico dei suoi ordini e modificare la propria password.
 2. L'utente non registrato può effettuare la registrazione al sistema.
 3. L'utente Gestore ordine può effettuare l'accesso al sistema utilizzando le proprie credenziali e può effettuare il logout.
 4. L'utente Gestore prodotti può effettuare l'accesso al sistema utilizzando le proprie credenziali e può effettuare il logout.
 5. L'utente Gestore ruoli può effettuare l'accesso al sistema utilizzando le proprie credenziali e può effettuare il logout.
 6. Permette al Gestore ruoli di aggiungere o rimuovere un dipendente (Gestore ordine, Gestore prodotti e Gestore ruoli).
- **GestioneCarrello:**
 1. Permette all'utente non registrato di visualizzare, aggiungere e rimuovere un prodotto al carrello.
 2. Permette al cliente di visualizzare, aggiungere e rimuovere un prodotto al carrello. Consente inoltre di procedere all'ordine
- **GestioneAcquisti:**
 1. Permette al cliente di inserire i dati di spedizione, i dati della carta e di confermare l'acquisto.
 2. Permette al gestore ordini di visualizzare, modificare, eliminare o confermare un ordine.
- **GestioneProdotti:**
 1. Il cliente può visualizzare i prodotti presenti nel sito.
 2. L'utente non registrato può visualizzare i prodotti presenti nel sito.
 3. Il Gestore prodotti può visualizzare, inserire, rimuove e modificare un prodotto.

2.3 Hardware/software mapping

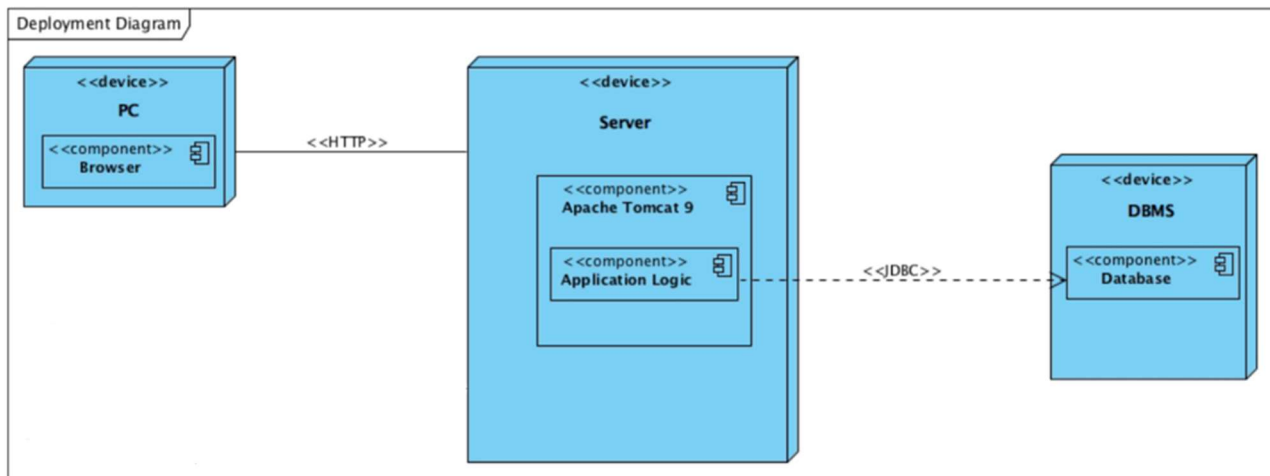
Gametop ha un'architettura di tipo client/server.

Il server riceve le richieste dal client e risponde in tempo utile.

I motivi di questa scelta sono:

- **Portabilità:** il sistema potrà essere utilizzato su una varietà di macchine e sistemi operativi, da computer fissi a dispositivi mobili;
- **Flessibilità:** il sistema presenterà modifiche all'interfaccia in base al tipo di utente che utilizza il sito per permettere ai vari attori di eseguire le funzionalità a loro dedicate in modo intuitivo

- **Affidabilità:** entrambi i componenti client e server devono essere affidabili ed essere in grado di mantenere i propri dati anche in seguito a guasti; quindi, deve essere possibile effettuare dei backup periodici al database con cadenza trimestrale.



Web Server

Il server utilizzato è Apache Tomcat.

Interface layer

L'utente utilizza il sistema mediante un Browser installato all'interno del suo dispositivo (ad es. Chrome, Firefox, Edge, Opera ecc...).

Application Logic layer

Le funzionalità del sistema sono implementate in linguaggio Java (Servlet).

Il codice in JavaServlet verrà tradotto in linguaggio HTML e JSP, inoltre, il codice risultante viene inviato al browser del client.

Database Server

Viene utilizzato il DBMS MySQL, il quale presenta molte API che permettono l'interazione tra sistema e database.

2.4 Gestione dei dati persistenti

Per la gestione dei dati persistenti si è deciso di utilizzare MySQL, questa scelta è stata fatta per ridurre i costi essendo MySQL gratuito ed anche perché i dati e le operazioni su di essi possono essere trattati in maniera efficace usando MySQL.

Le seguenti entità, provenienti dal Class Diagram verranno rese persistenti:

- Prodotti
- Cliente
- Carrello
- Ordini
- Personale

2.5 Controllo degli accessi e sicurezza

Gli accessi vengono gestiti tramite un sistema di autenticazione che prevede l'inserimento di email e password riservate per ogni utente.

L'accesso alle diverse funzionalità ed aree riservate del sistema dipende strettamente dal tipo di ruolo che ha l'utente registrato con determinate credenziali.

La sessione del server verrà usata per tenere traccia dell'utente loggato.

Per questioni di efficienza, la sessione sarà attiva per 10 minuti dopo l'ultima interazione dell'utente col sistema. Il salvataggio delle password nel database sarà criptato.

Di seguito viene elencata la Global Access Table che descrive per ogni cella della matrice quali operazioni può effettuare ogni attore su un oggetto.

	Account	Prodotti	Carrello	Ordini	Acquisti	Ruoli
Cliente non registrato	- Registrazione	-Visualizzare elenco prodotti -Visualizzare dettagli prodotto	-Aggiungere prodotto al carrello - Rimuovere prodotto dal carrello			
Cliente registrato	-Login -Modificare password -Visualizzare storici ordini effettuati -Logout	-Visualizzare prodotti -Visualizzare dettagli prodotto	-Aggiungere prodotto al carrello -Rimuovere prodotto dal carrello -Procedere all'ordine	-Iniziare il processo di creazione di un ordine	-Inserisce dati di spedizione -Inserisci dati per il pagamento -Conferma pagamento	
Gestore Ordini	-Login -Logout			- Visualizzare tutti gli ordini - Confermare ordine -Modificare ordine		
Gestore prodotti	-Login -Logout	-Aggiungi prodotto -Rimuovi prodotto -Modifica prodotto				

Gestore ruoli		-Visualizza prodotti				
	-Login -Logout					-Aggiunta nuovo gestore ordini -Aggiunta nuovo gestore prodotti -Aggiunta nuovo gestore ruoli -Visualizza dipendenti

2.6 Global software control

Il sistema ha un controllo software globale di tipo event-driven.

Il controllo è gestito tramite un dispatcher che si occupa di ricevere eventi(richieste dell'utente) e chiama le funzioni necessarie mediante callback.

Il Web Server si occupa di gestire le richieste effettuate dagli utenti (client).

Il server affida le richieste agli oggetti Control (le Java Servlet), che gestiranno la richiesta eventualmente interagendo con i Model.

Il Web Server aggiornerà le View coerentemente alle operazioni fatte e che saranno visualizzate dal client attraverso la creazione di codice HTML dalla pagina JSP.

2.7 Boundary conditions

Le condizioni limite riguardano l'inizializzazione, la terminazione e le failure del sistema.

-Inizializzazione del sistema:

Identificativo: Start Server

Attori: Amministratore

Flusso di eventi:

- 1) L'amministratore decide di avviare il server, accede alla macchina Server e preme su 'Start'
- 2) Il sistema, con le opportune procedure di avvio, attiva il server e i relativi servizi rendendosi disponibile ad eventuali richieste.
- 3) Il sistema notifica il successo della procedura.

Condizione di uscita: L'amministratore visualizza la notifica di successo

-Terminazione del sistema:

Identificativo: Stop Server

Attori: Amministratore

Flusso di eventi:

- 1) L'amministratore decide di arrestare il server, accede alla macchina Server e preme su 'Stop'
- 2) Il sistema riceve la richiesta, effettua le operazioni ancora in corso e spegne il server

Condizione di uscita: L'amministratore visualizza la notifica di successo.

-Fallimenti del sistema:

Nel caso di fallimento hardware/software, si procede ad un riavvio del sistema.

Per quanto riguarda invece la gestione dei dati persistenti, si procederà ad un backup periodico per minimizzare il rischio di perdita dovuto a guasti al DBMS.

Identificativo: Restart Server

Attori: Amministratore

Flusso di eventi:

- 1) L'amministratore decide di riavviare il server, accede alla macchina Server e preme su 'Reset'
- 2) Il sistema riceve la richiesta, effettua le operazioni ancora in corso e riavvia il server

Condizione di uscita: L'amministratore visualizza la notifica di successo.

3 Subsystem service

Di seguito verranno descritti I servizi offerti dai vari sottosistemi, con relativa descrizione.

Sottosistema	Descrizione
Account	Consente ad un Utente non registrato di effettuare la registrazione e a un Utente registrato di effettuare il login. Permette l'inserimento e la rimozione del personale (Gestore ordine, Gestore prodotti e Gestore ruoli) compreso login e logout.

Servizio	Descrizione
Login	Consente ad un Utente di accedere al sistema tramite username e password
Logout	Consente ad un Utente registrato, Gestore ordini, Gestore prodotti e Gestore ruoli di terminare l'autenticazione al sistema
Login Gestore ordine	Consente ad un Gestore Ordini di accedere alle proprie funzionalità
Registrazione Utente	Permette ad un Utente non registrato di creare un account nel sistema
Login Gestore Prodotti	Consente ad un Gestore prodotti di accedere alle proprie funzionalità
Modifica password Utente	Consente ad un Utente registrato di modificare la password relativa al proprio account
Modifica dati personale (Gestore ordine e prodotto)	Consente ad un Gestore ruoli di modificare le informazioni relative all'account di un personale.
Login Gestore ruoli	Consente ad un Gestore ruoli di accedere alle proprie funzionalità
Registrazione personale	Consente ad un Gestore ruoli di inserire le informazioni relative all'account di un personale
Rimuovi personale	Consente ad un Gestore ruoli di rimuovere le informazioni relative all'account di un personale
Visualizzazione storico ordini	Questa funzionalità permette al cliente loggato di visualizzare lo storico degli ordini effettuati.

Sottosistema	Descrizione
Carrello	Permette di visualizzare il carrello, aggiungere e rimuovere un prodotti da quest'ultimo.

Servizio	Descrizione
Visualizza carrello	Consente di visualizzare i prodotti aggiunti nel carrello e il prezzo totale
Aggiungi prodotto al carrello	Consente di aggiungere un prodotto al carrello
Rimuovi Prodotto dal carrello	Consente di rimuovere un prodotto dal carrello
Procedere All'acquisto	Permette ad un Utente registrato di iniziare la procedura per completare l'acquisto di un prodotto. Dopo tale operazione il carrello verrà svuotato

Sottosistema	Descrizione
Prodotto	Permette l'aggiunta, la modifica e la rimozione di un prodotto da parte di un Gestore prodotti. Consente inoltre la visualizzazione di tutto il catalogo dei prodotti

Servizio	Descrizione
Aggiunta prodotto	Permette ad un Gestore prodotti di inserire un nuovo prodotto alla categoria selezionata
Rimuovi prodotto	Permette ad un Gestore prodotti di rimuovere un prodotto.
Modifica prodotto	Permette ad un Gestore prodotti di modificare le informazioni relative ad un prodotto precedentemente inserito
Visualizzare Prodotti	Permette di visualizzare i prodotti presenti nel sito
Visualizzare Prodotti in base alla categoria	Permette ad un utente(registrato e non)di visualizzare i prodotti della categoria selezionata

Sottosistema	Descrizione
Acquisti	Consente ad un Utente registrato di inserire i dati di spedizione e i dati di pagamento per completare l'acquisto, generando un ordine. Permette la conferma degli ordini effettuati dai clienti. Consente inoltre la visualizzazione di tutti gli ordini ricevuti

Servizio	Descrizione
Inserisci dati spedizione	Permette di inserire i dati di spedizione con le relative informazioni del destinatario.
Inserisci dati pagamento	Permette di inserire i dati della carta di credito, necessari per effettuare il checkout
Pagare/genera ordine	Permette ad un utente registrato di acquistare i prodotti inseriti nel carrello. Dopo tale operazione il carrello verrà svuotato e genererà l'ordine.
Confermare ordine	Permette ad un Gestore ordine di poter confermare un ordine.
Modifica/elimina ordine	Permette ad un Gestore ordine di modificare le informazioni relative ad un ordine se non ancora spedito/confermato. Permette ad un Gestore ordini di eliminare un ordine
Visualizzare ordini	Permette di visualizzare tutti gli ordini ricevuti non ancora confermati.