



System Design Document

[Versione 2]



Sommario

Informazioni sul documento	4
Generalità.....	4
Team Project.....	4
Revision History	4
1. Introduzione	6
1.1 Obiettivo del sistema	6
1.2 Obiettivi di design (Design goals)	7
1.3 Definizioni, acronimi e abbreviazioni	9
1.4 Riferimenti.....	10
1.5 Organizzazione del documento.....	10
2. Architettura del sistema software corrente	11
3. Architettura del sistema software proposto	12
3.1 Panoramica sulla sezione	12
3.2 Decomposizione in sottosistemi	13
3.3 Mapping hardware/software	20
3.4 Gestione dei dati persistenti	22
3.5 Controllo degli accessi e sicurezza	30
3.6 Controllo globale del software	31
3.7 Boundary conditions.....	31
3.8 Servizi dei sottosistemi	34
4. Glossario	39





Informazioni sul documento

Generalità

- **Progetto:** TechHeaven
- **Versione:** [Versione 2]
- **Documento:** Documento di design del sistema
- **Data:** [03/11/2024]

Team Project

Nome Membro	Matricola	Ruolo	Contatti
Dorotea Serrelli	0512113740	Project manager	d.serrelli1@studenti.unisa.it
Raffaella Sabatino	0512115114	Team member	r.sabatino17@studenti.unisa.it

Revision History

Data	Versione	Descrizione	Autore
23/01/2024	0.1	Scrittura delle sezioni "Introduzione" e "Architettura del sistema software corrente"	Tutto il team
25/01/2024	0.2	Scrittura dei paragrafi "Decomposizione in sottosistemi" e "Mapping hardware/software" della sezione "Architettura del sistema software proposto"	Tutto il team



31/01/2024	0.3	Scrittura paragrafo "Gestione dei dati persistenti"	Dorotea Serrelli
08/02/2024	0.4	Scrittura sezione "Controllo globale del sistema"	Tutto il team
18/02/2024	0.5	Scrittura sezione "Boundary conditions"	Raffaella Sabatino
22/02/2024	0.6	Modifica della matrice del controllo degli accessi	Tutto il team
29/02/2024	0.7	Definizione dei servizi dei sottosistemi	Tutto il team
02/03/2024	0.8	Revisione dei servizi dei sottosistemi e definizione del glossario	Tutto il team
31/07/2021	0.9	Revisione documento	Tutto il team
31/07/2024	1.0	Rilascio prima versione SDD	Dorotea Serrelli
03/11/2024	1.1	Revisione documento e rilascio seconda versione SDD	Tutto il team



1. Introduzione

1.1 Obiettivo del sistema

La società TechHeavenSrl è responsabile dall'anno 2000 della gestione del negozio "TechHeaven – Il paradiso digitale", specializzato nella vendita di prodotti elettronici, elettrodomestici, telefonia.

Tale negozio è, attualmente, un punto vendita di riferimento nella zona per lo smercio, la qualità e il prezzo dei prodotti, al punto che soddisfa un grande bacino di utenza, quasi esclusivamente residente nella provincia.

La società intende espandere i confini della propria attività ed ampliare la clientela, avvalendosi di un sistema software che consenta, sotto il profilo soggettivo, una maggiore conoscibilità della società e dell'affidabilità della stessa; sotto il profilo oggettivo, favorisca l'incremento della vendita dei prodotti.

Il sistema software verrà sviluppato per fornire alla clientela informazioni sulla società, sul punto vendita e sui prodotti trattati.

La piattaforma permetterà, infatti, al cliente di registrarsi, in modo da poter visionare i prodotti in vendita ed acquistarli, tenere traccia dello stato degli ordini effettuati presso il negozio online e creare una lista di prodotti desiderati (wishlist).

La piattaforma, inoltre, consentirà l'accesso ai seguenti dipendenti:

- Gestore degli ordini: responsabile del processo di acquisizione, registrazione ed evasione degli ordini dei clienti, nonché dell'elaborazione di richieste di approvvigionamento di prodotti da inoltrare, poi, all'ufficio acquisti.
- Gestore del catalogo: responsabile della presentazione, organizzazione e gestione del catalogo dei prodotti venduti dal negozio.

La piattaforma, quindi, consentirà al gestore degli ordini di visionare gli ordini commissionati dai clienti al negozio e gli ordini che sono stati spediti, nonché preparare un ordine alla spedizione e fare richiesta di approvvigionamento di prodotti mancanti.

Tale software, inoltre, permetterà al gestore del catalogo di visionare il catalogo e di poter inserire, cancellare e modificare un prodotto nel catalogo.

1.2 Obiettivi di design (Design goals)

In questa sezione sono riportati i Design Goals, ovvero le caratteristiche qualitative che deve possedere il sistema software. La formalizzazione esplicita di tali caratteristiche sarà costantemente utilizzata come linea guida per le scelte progettuali che si affronteranno nelle attività successive.

In base alle linee guida delineate dagli autori del libro *Object Oriented Software Engineering*, Bernd Bruegge e Allen H. Dutoit, i design goal sono stati raggruppati nelle seguenti categorie:

- Performance: includono i requisiti di spazio e velocità imposti sul sistema.
- Dependability: determinano quanto sforzo deve essere speso per minimizzare i fallimenti del sistema (crash, falle di sicurezza) e le loro conseguenze.
- Maintenance: determina quanto sforzo è necessario per modificare il sistema dopo il suo rilascio.
- End User: includono qualità del sistema che sono desiderabili dal punto di vista dell'utente, ma che non sono state coperte dai criteri di Performance e Dependability.

Nella seguente tabella si riportano le informazioni dei design goals individuati nel RAD : il nome, la descrizione, la categoria di appartenenza e il requisito non funzionale da cui discende.

Nome	Descrizione	Categoria	RNF di origine
DG_1 Quantità dei dati	Il sistema dovrà garantire un'alta capacità di memorizzazione per poter conservare tutte le informazioni necessarie per l'erogazione dei servizi che offre.	Performance	RNF_11
DG_2 Gestione permessi sistema	Il sistema deve garantire la separazione netta delle operazioni, sulla base degli utenti che possono effettuarle.	Dependability	RNF_3

DG_3 Affidabilità	Il sistema deve garantire che tutte le operazioni avvengano con successo.	Dependability	RNF_4
DG_4 Robustezza	Il sistema deve essere in grado di gestire l'inserimento di dati non validi da parte dell'utente, generando messaggi di errore.	Dependability	RNF_5
DG_5 Crittografia password utente	Il sistema deve memorizzare la password di accesso di un utente non in chiaro, ossia applicare su di essa un algoritmo crittografico.	Dependability	RNF_6
DG_6 Protezione da attacchi SQLInjection	Il sistema deve essere protetto da attacchi di tipo SQLInjection.	Dependability	RNF_7
DG_7 Protezione da Cross-site scripting	Il sistema deve difendersi da attacchi di tipo Cross-site scripting.	Dependability	RNF_8
DG_8 Protocollo HTTPS	Il sistema deve proteggere l'integrità e la riservatezza dei dati scambiati tra client e server, mediante l'uso di un certificato SSL/TLS.	Dependability	RNF_9
DG_9 Portabilità	Il sistema sarà disponibile tramite pagina web da qualsiasi dispositivo (tablet, smartphone, computer, ...). Ciò rende il sistema altamente portabile in quanto non è necessario adattare il sistema tra diversi sistemi operativi.	Dependability	RNF_18 RNF_14
DG_10 Manutenibilità	Il sistema dovrà essere sviluppato seguendo i	Maintenance	RNF_12

	principali standards per la buona manutenibilità.		
DG_11 Estendibilità	Il sistema dovrà essere sviluppato seguendo i principali standards che garantiscono una buona estendibilità delle funzionalità.	Maintenance	RNF_13
DG_12 Facilità di utilizzo	Il sistema deve essere facilmente utilizzabile anche da un utente non esperto.	End user	RNF_1
DG_13 Interfaccia intuitiva	L'interfaccia utente della piattaforma deve permettere di eseguire azioni in modo chiaro e semplice, rendendo ben esplicita la funzionalità di ogni elemento visuale.	End user	RNF_2
DG_14 Sistema responsive	Il sistema dovrà offrire un'interfaccia grafica utente che sia in grado di adattarsi, principalmente, allo schermo di uno dei seguenti dispositivi: smartphone, tablet e computer.	End user	RNF_10

1.3 Definizioni, acronimi e abbreviazioni

In questa sezione si riportano di seguito l'insieme delle definizioni, degli acronimi e delle abbreviazioni che verranno utilizzati in questo documento:

- Design goals: le caratteristiche qualitative che il sistema software deve possedere, ottenute dal dominio applicativo e dal cliente/dagli utenti finali;
- Sottosistema: una parte sostituibile del sistema software che offre un sottoinsieme di servizi che soddisfano una o più funzionalità definite nel RAD;
- Dati persistenti: i dati che sopravvivono alla singola esecuzione del sistema e che, dunque, devono essere memorizzati;



- Mapping hardware/software: lo studio della connessione tra le componenti logiche e le componenti fisiche che costituiscono il sistema;
- Boundary conditions: le modalità di avvio, spegnimento e configurazione del sistema;
- RAD: Requirements Analysis Document;
- SDD: System Design Document.

1.4 Riferimenti

Di seguito si elencano i documenti utili alla comprensione del documento:

- Problem statement;
- RAD;
- Database Design Document (DDD);
- Object Design Document (ODD).

1.5 Organizzazione del documento

Il presente documento consta di quattro sezioni:

1. **Introduzione:** si descrivono lo scopo del sistema, gli obiettivi di design su cui il sistema deve essere sviluppato, le definizioni e gli acronimi utilizzati nel documento, nonché i riferimenti agli altri documenti rilasciati.
2. **Architettura del sistema software corrente:** si delinea lo stato attuale dell'architettura del software presente.
3. **Architettura del sistema software proposto:** si illustra come il sistema verrà decomposto in sottosistemi, il mapping Hardware/Software dei sottosistemi, la gestione dei dati persistenti, le politiche di sicurezza e di gestione degli accessi adottate e il flusso di controllo del sistema; inoltre, verranno presentate la struttura dei singoli sottosistemi e le boundary conditions riguardanti l'intero sistema.
4. **Glossario:** riporta la lista dei termini usati nel documento con annessa spiegazione.



2. Architettura del sistema software corrente

La società, al momento, non presenta un sistema informatico in grado di raccogliere gli ordini dei clienti in modo automatico, bensì prende in carico solo gli ordini effettuati dai clienti che risiedono nella regione in cui è localizzato il negozio.

Attualmente, gli ordini richiesti dai clienti vengono inviati al negozio tramite telefono o fax.

Come si evince dal contesto appena descritto, questa modalità di gestione degli ordini comporta perdita di tempo per la raccolta dei dati ed oneri aggiuntivi per la relativa gestione, a discapito della produttività dell'azienda.



3. Architettura del sistema software proposto

3.1 Panoramica sulla sezione

In questa sezione si descrivono le scelte progettuali adottate e le modalità con cui sono state condotte le seguenti attività di progettazione del sistema:

- Decomposizione del sistema;
- Mapping hardware/software;
- Gestione dei dati persistenti;
- Gestione della sicurezza e del controllo degli accessi;
- Flusso di controllo globale del sistema;
- Boundary conditions.

Sono stati considerati diversi stili architetturali per il sistema software proposto (client-server, peer-to-peer, three-tier, repository, ...) e, tra questi, si è scelto di optare per lo stile Three-tier.

Il motivo di tale scelta risiede nel fatto che lo stile architetturale Three-tier è adatto per i sistemi interattivi, nei quali è richiesta la gestione di interazioni complesse tra i componenti del sistema e l'utente.

Questo stile architetturale, inoltre, prevede la separazione della logica di presentazione da quella relativa all'elaborazione ed archiviazione dei dati, caratteristica che offre una serie di vantaggi:

- Leggibilità;
- Manutenzione;
- Riutilizzo.

Nello sviluppo del sistema si utilizzeranno come componenti off-the-shelf:

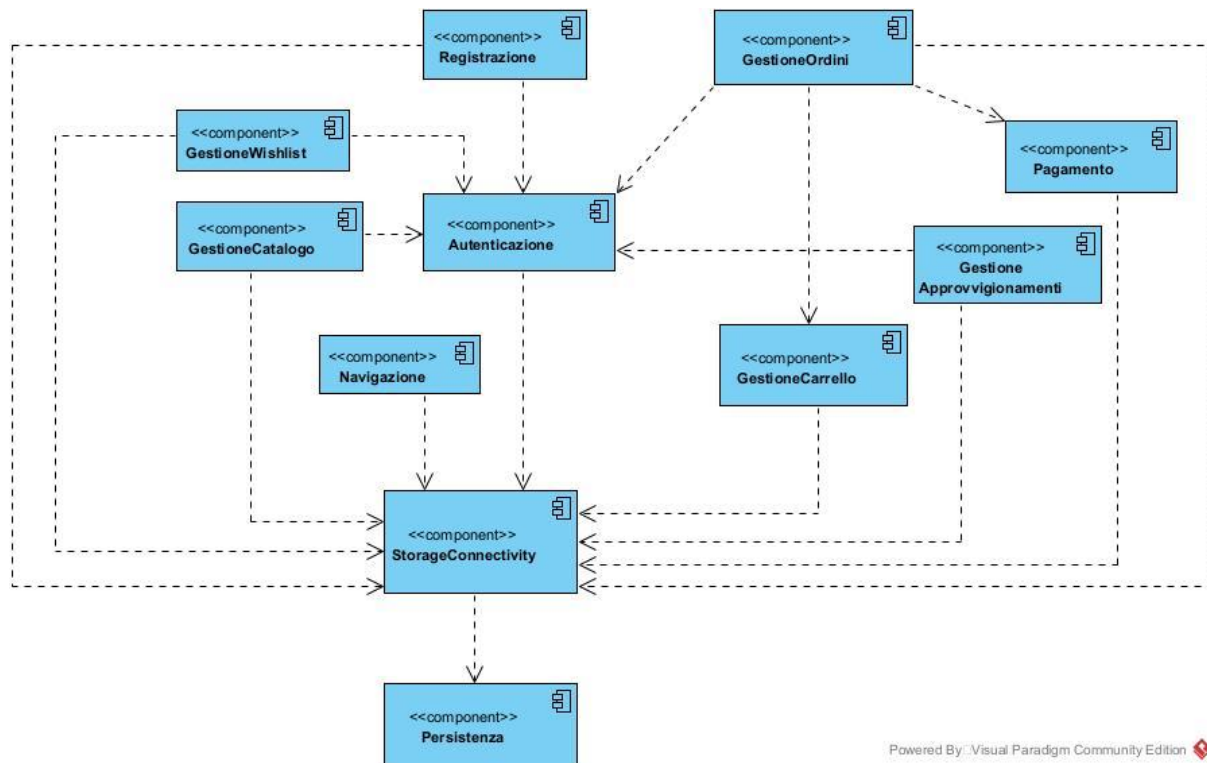
- per la presentazione dei dati e l'interfaccia utente HTML, CSS e Javascript;
- per la logica applicativa e, quindi, il back-end si utilizzerà Java e JSP (JavaServer Pages);
- per la gestione dei dati persistenti, si è scelto di utilizzare un database relazionale con DBMS MySQL, che utilizza il linguaggio SQL per gestire e interrogare i dati. La scelta è motivata dai vantaggi dell'uso di un database: scalabilità ed offerta dei servizi per la gestione della concorrenza, il controllo degli accessi, il recupero da arresto.

3.2 Decomposizione in sottosistemi

Analizzando il modello dei casi d'uso ed il modello dinamico presenti nel RAD, si sono individuati i seguenti sottosistemi:

- Registrazione : si occupa di gestire la registrazione dei nuovi clienti del negozio;
- Autenticazione: è responsabile dell'autenticazione di un cliente o di un dipendente del negozio (fino adesso gestore degli ordini e gestore del catalogo) nel sistema, dell'accesso all'area riservata e modifica dei dati personali dell'utente, nonché della reimpostazione della password associata all'utente;
- GestioneWishlist : si occupa della visualizzazione della lista dei desideri del cliente, dell'aggiunta di prodotti e della rimozione degli stessi dalla wishlist;
- GestioneCarrello: è responsabile della visualizzazione del carrello virtuale e delle operazioni di aggiunta, rimozione e variazione della quantità di un prodotto del carrello;
- Navigazione: si occupa della ricerca di un prodotto nel catalogo per barra di ricerca o per menù di navigazione, della visualizzazione della pagina di risultati della ricerca e della visualizzazione delle referenze di un prodotto;
- GestioneOrdini: offre servizi relativi alla creazione e preparazione di un ordine del cliente alla spedizione, della visualizzazione degli ordini da spedire e degli ordini evasi dal negozio;
- GestioneApprovvigionamenti: si occupa della visualizzazione ed elaborazione di richieste di approvvigionamento di prodotti fatte dal gestore degli ordini;
- Pagamento: si occupa di gestire diverse modalità di pagamento dei prodotti presenti in un ordine del cliente in fase di acquisto (fino adesso pagamento con carta di credito, in contrassegno, Paypal);
- GestioneCatalogo: è responsabile della gestione del catalogo detenuta dal gestore del catalogo (visualizzazione del catalogo, aggiunta di un prodotto, rimozione di un prodotto, modifica delle specifiche di un prodotto);
- Persistenza: si occupa di gestire la persistenza dei dati con un database;
- Storage Connectivity: si interpone tra i vari sottosistemi e il sottosistema "Persistenza".

Di seguito si riporta in UML il component diagram relativo alla decomposizione del sistema software:

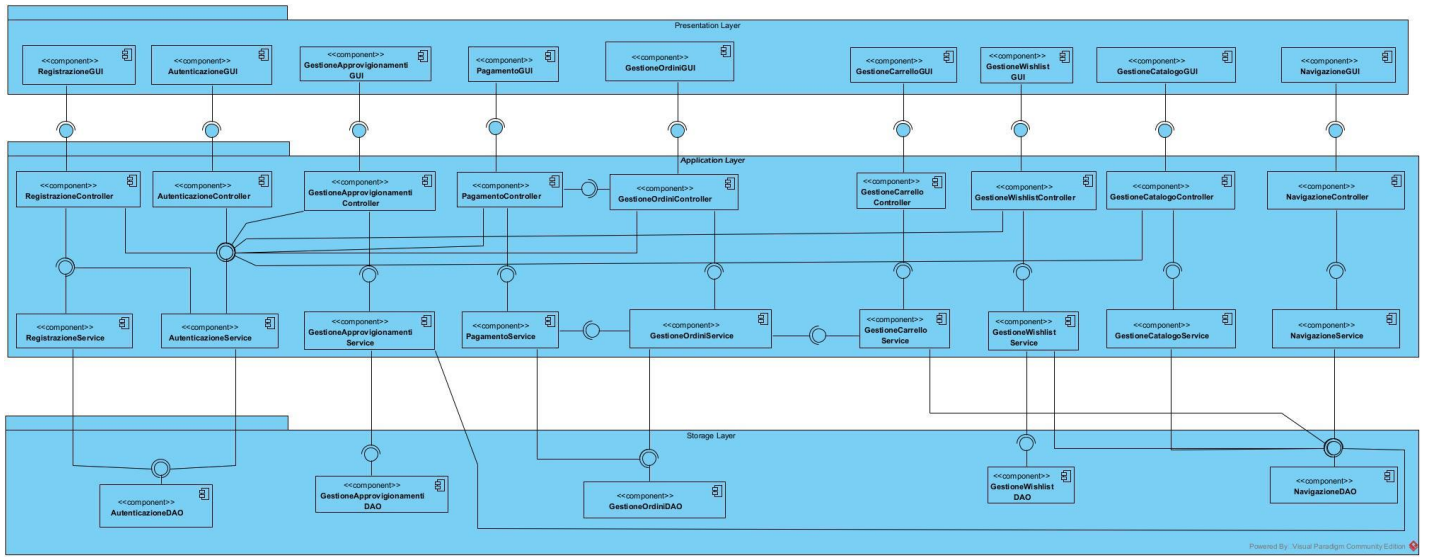


Alcuni sottosistemi saranno gestiti dalle seguenti componenti COTS (Commercial off-the-shelf):

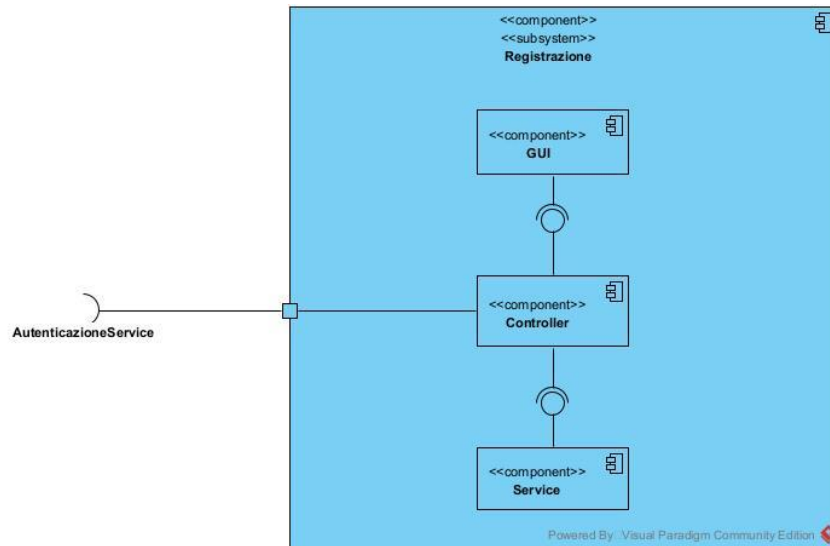
- Persistenza sarà gestita attraverso un DBMS relazionale su sistema MySQL.

Si riporta, inoltre, il diagramma architetturale del sistema software, offrendo una vista dettagliata di ciascun sottosistema, evidenziando le componenti principali:

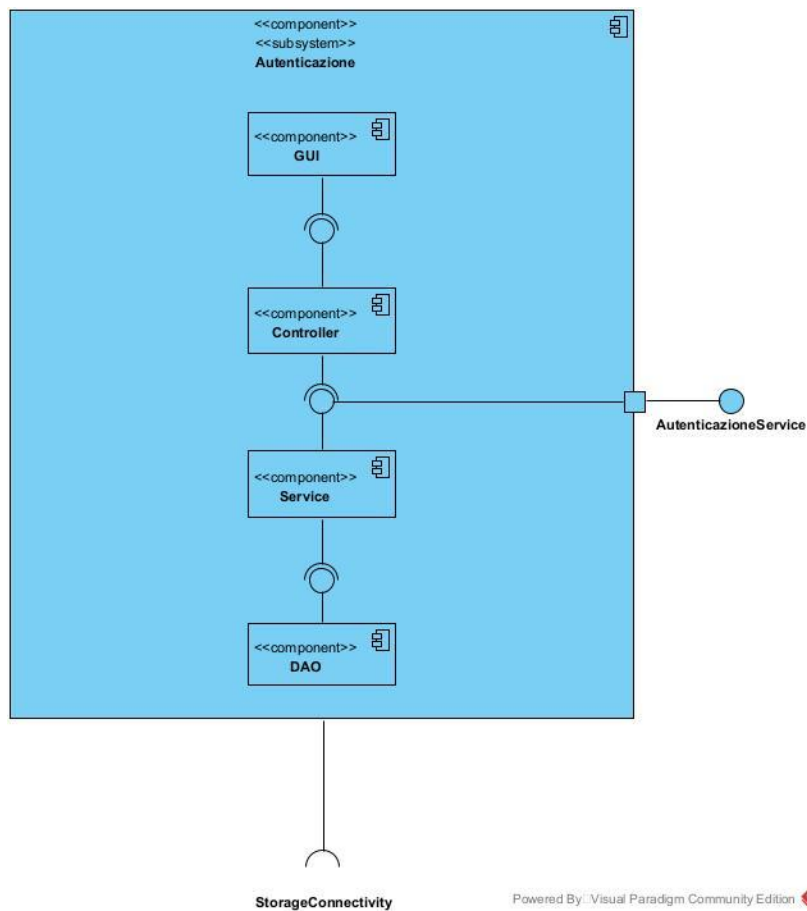
- GUI (Graphic User Interface): contiene le varie interfacce grafiche che saranno renderizzate per creare le pagine web da mostrare al cliente.
- Controller: si occupa della logica per il flusso di controllo del sistema.
- Service: si occupa della logica di business.
- DAO: Data Access Object, che si occupa di fornire accesso ai dati persistenti.



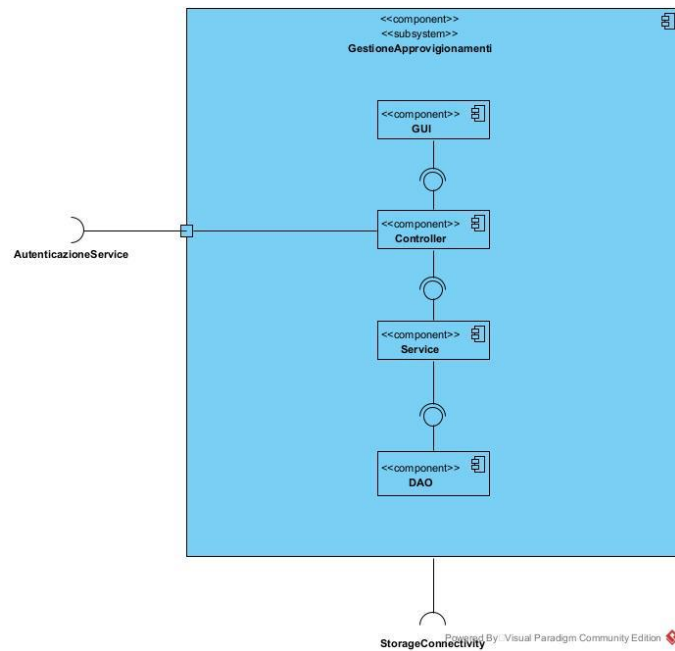
Sottosistema Registrazione



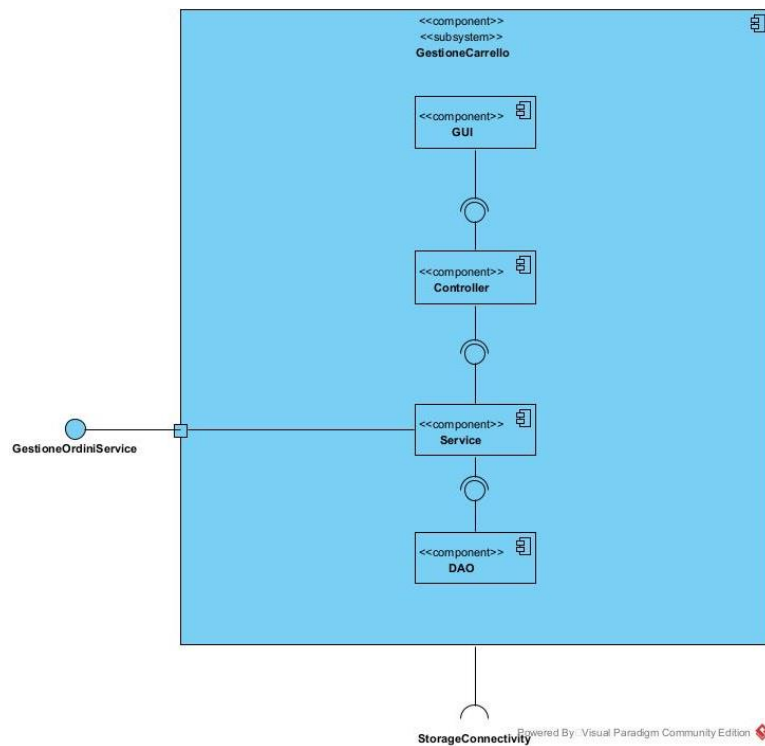
Sottosistema Autenticazione



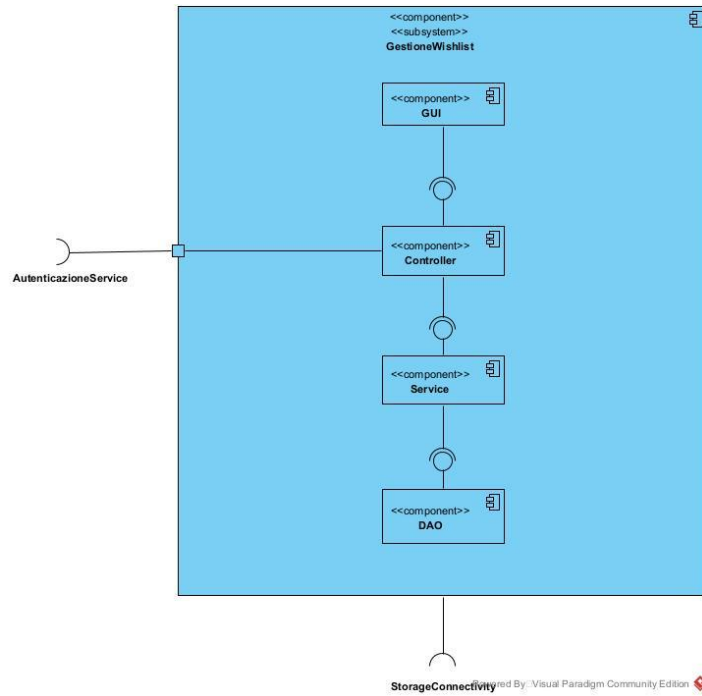
Sottosistema Gestione approvvigionamenti



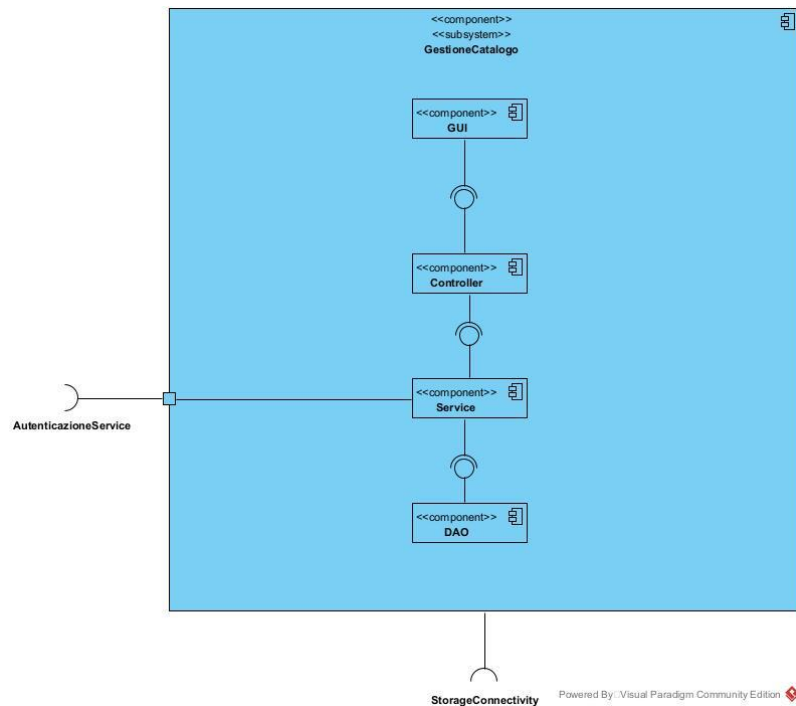
Sottosistema Gestione carrello



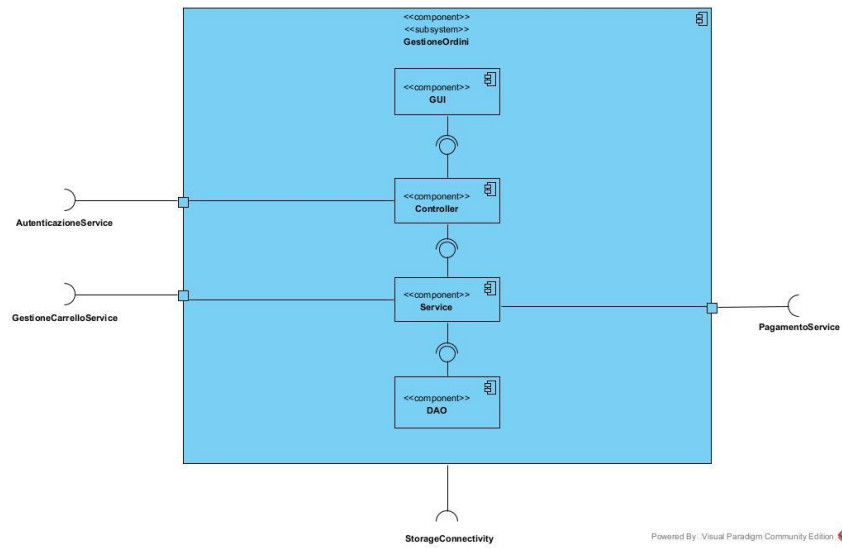
Sottosistema GestioneWishlist



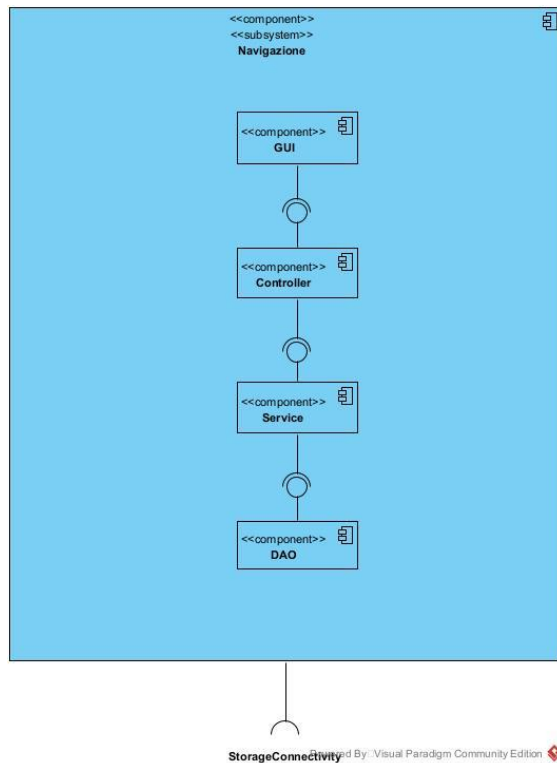
Sottosistema Gestione Catalogo



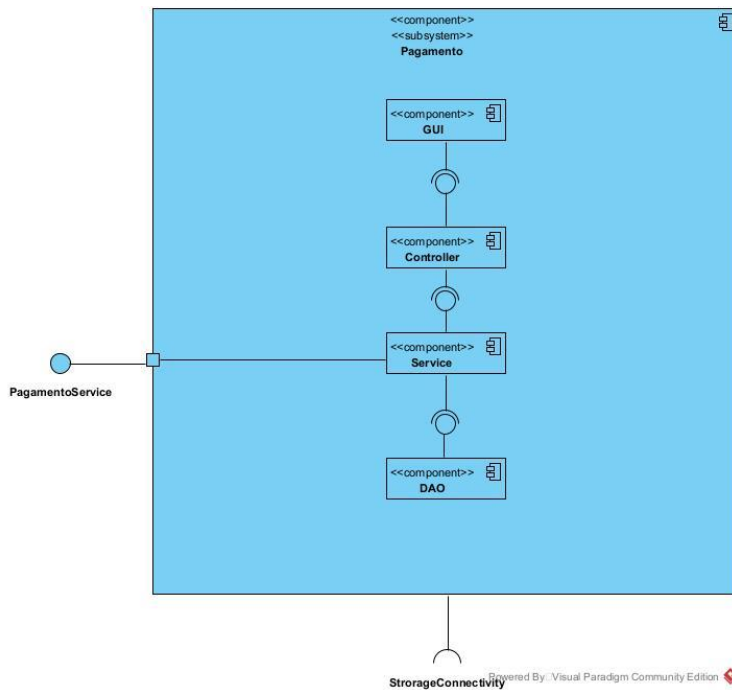
Sottosistema Gestione degli ordini



Sottosistema Navigazione



Sottosistema Pagamento

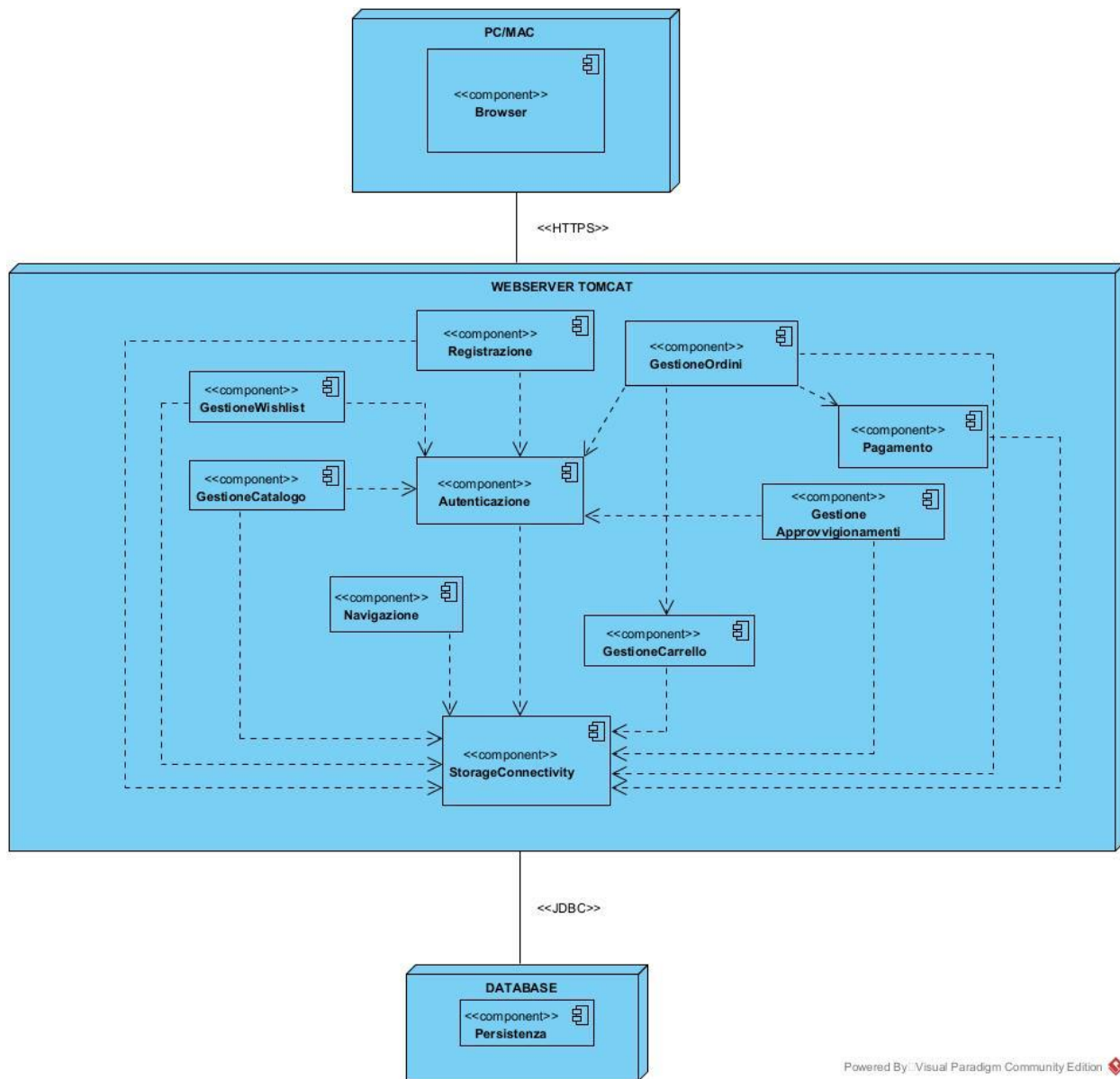


3.3 Mapping hardware/software

L' applicazione web che verrà sviluppata si basa su una piattaforma hardware costituita da un server che risponde alle richieste effettuate dai clienti; tali clienti effettueranno le richieste da una qualsiasi macchina con un browser ed una connessione ad Internet.

Visto che il nostro sistema è un' applicazione Web-based, residente su un web server basata su un'architettura non distribuita, risiede su un solo nodo.

Si illustra di seguito un UML deployment diagram che descrive il mapping hardware/software.





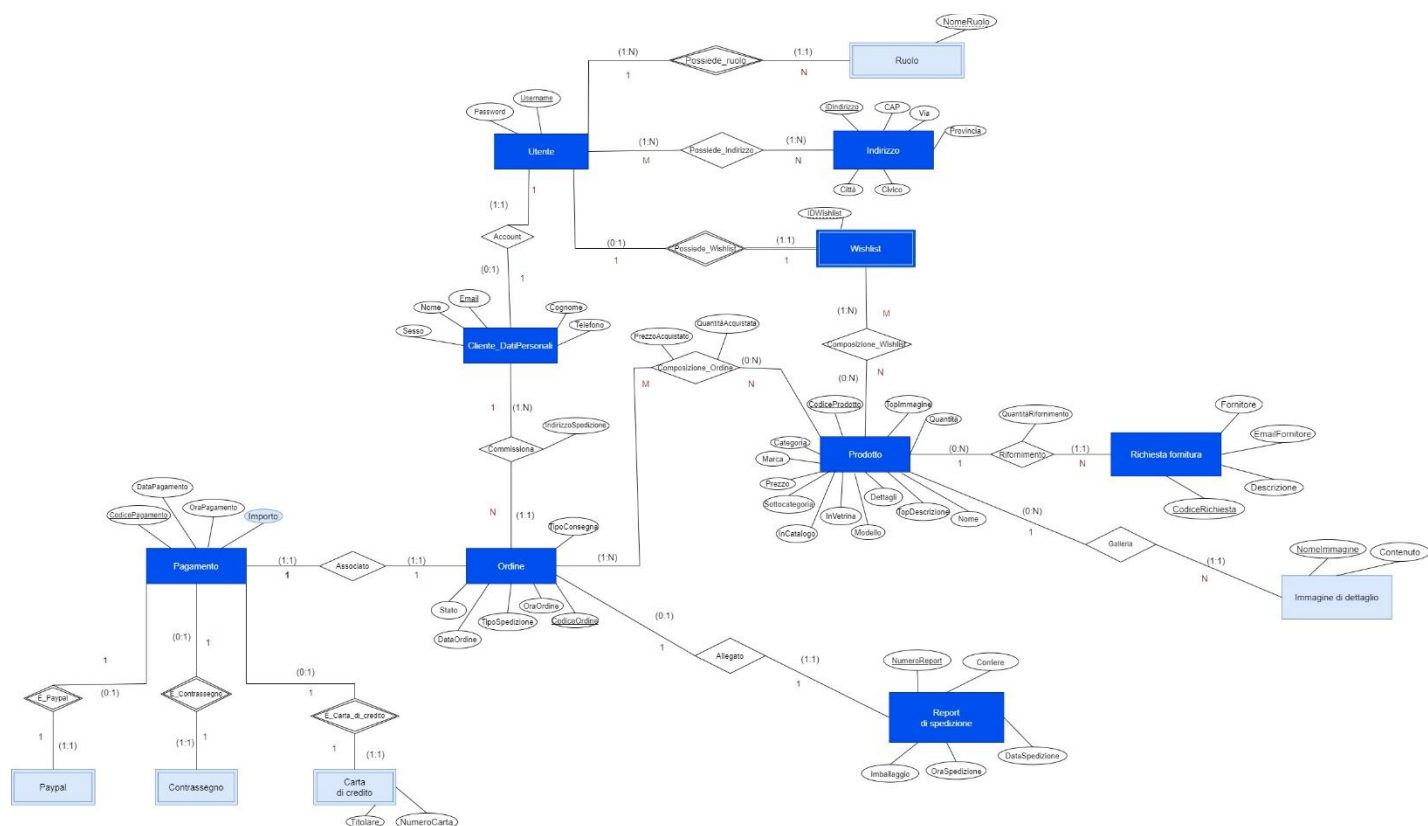
3.4 Gestione dei dati persistenti

Per la memorizzazione dei dati persistenti del sistema si è deciso di utilizzare un database relazionale. Esso consente di gestire agevolmente l'accesso concorrente ai dati e, allo stesso tempo, garantire la consistenza dei dati tramite l'utilizzo di un DBMS.

La scelta di utilizzare un DBMS è guidata dalla volontà di rimanere coerenti con i design goals stabiliti, potendo contare su:

- **Imposizioni di vincoli di integrità sui dati:** un DBMS permette di specificare diversi tipi di vincoli per garantire l'integrità dei dati e, poi, ad ogni cambiamento di stato del database, controlla che tali vincoli siano soddisfatti.
- **Privatezza dei dati:** un DBMS permette un accesso protetto ai dati. Utenti diversi (nel nostro caso il cliente, il gestore degli ordini e il gestore del catalogo) possono avere accesso ad apposite porzioni del database e sono abilitati ad eseguire determinate operazioni su di esse.
- **Affidabilità dei dati:** un DBMS offre dei metodi per salvare copie dei dati e per ripristinare lo stato della base di dati in caso di guasti software e hardware.
- **Atomicità delle operazioni:** un DBMS permette di effettuare sequenze di operazioni in modo atomico, ovvero o l'intera sequenza di operazioni viene eseguita con successo oppure nessuna di queste operazioni viene effettuata. L'atomicità delle transazioni permette di mantenere uno stato del database consistente con la realtà modellata.

Si riporta lo schema EER per la progettazione concettuale del database:



Di seguito si descrive il glossario dei dati: le entità memorizzate nel database e, per ognuna, gli attributi individuati e relativi domini di valori.

Nome Entità			
Cliente_DatiPersonali			
Descrizione	Le informazioni personali inserite dal cliente per registrarsi al sistema (nome, cognome, sesso, numero di telefono, email)		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
email	Varchar(50)	PRIMARY KEY	NOT NULL
nome	Varchar(40)		NOT NULL
cognome	Varchar(50)		NOT NULL
sesso	enum(M, F)		NOT NULL
telefono	char(12)		NOT NULL



<i>Nome Entità</i>	Utente		
<i>Descrizione</i>	Tiene traccia delle credenziali di accesso (username e password)		
<i>Nome campo</i>	<i>Tipo</i>	<i>Vincolo di chiave</i>	<i>Altri vincoli</i>
username	Varchar(25)	PRIMARY KEY	NOT NULL
userPassword	Varchar(50)		NOT NULL
email	Varchar(50)	FOREIGN KEY	NOT NULL

<i>Nome Entità</i>	Ruolo		
<i>Descrizione</i>	Memorizza i ruoli associati per ogni utente registrato nel sistema		
<i>Nome campo</i>	<i>Tipo</i>	<i>Vincolo di chiave</i>	<i>Altri vincoli</i>
nomeRuolo	Varchar(30)	PRIMARY KEY	NOT NULL
utente	Varchar(25)	PRIMARY KEY FOREIGN KEY	NOT NULL

<i>Nome Entità</i>	Indirizzo		
<i>Descrizione</i>	Le informazioni relative all'indirizzo di un cliente		
<i>Nome campo</i>	<i>Tipo</i>	<i>Vincolo di chiave</i>	<i>Altri vincoli</i>
idIndirizzo	int	PRIMARY KEY	NOT NULL AUTO_INCREMENT
via	Varchar(50)		NOT NULL
numCivico	Varchar(4)		NOT NULL
citta	Varchar(50)		NOT NULL
CAP	char(5)		NOT NULL
provincia	char(2)		NOT NULL



Nome Entità	Possiede_Indirizzo		
Descrizione	Tiene traccia dell'insieme di indirizzi associati a ciascun cliente		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
utente	Varchar(25)	PRIMARY KEY FOREIGN KEY	NOT NULL
indirizzo	int	PRIMARY KEY FOREIGN KEY	NOT NULL

Nome Entità	Wishlist		
Descrizione	La wishlist creata da un cliente		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
utente	Varchar(25)	PRIMARY KEY FOREIGN KEY	NOT NULL
idWishlist	int	PRIMARY KEY	NOT NULL

Nome Entità	Prodotto		
Descrizione	Le informazioni relative ad un prodotto venduto dal negozio: codice, nome, descrizione breve, descrizione dettagliata, prezzo, categoria, sottocategoria, marca, modello, quantità in magazzino, immagine di presentazione, inserito in una vetrina del negozio online, disponibile nel catalogo		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
CodiceProdotto	int	PRIMARY KEY	NOT NULL
Nome	Varchar(80)		NOT NULL
TopDescrizione	text		NOT NULL
Dettagli	text		
Prezzo	float		NOT NULL
Categoria	Varchar(30)		NOT NULL
Sottocategoria	Varchar(50)		
Marca	Varchar(50)		NOT NULL
Modello	Varchar(40)		
InCatalogo	boolean		NOT NULL



InVetrina	boolean		NOT NULL
TopImmagine	mediumblob		
Quantità	int		NOT NULL

Nome Entità	Composizione_Wishlist		
Descrizione	L'insieme dei prodotti desiderati da un cliente che ha salvato nella wishlist		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
utente	Varchar(25)	PRIMARY KEY FOREIGN KEY	NOT NULL
wishlist	int	PRIMARY KEY FOREIGN KEY	NOT NULL
prodotto	int	PRIMARY KEY FOREIGN KEY	NOT NULL

Nome Entità	Richiesta_fornitura		
Descrizione	Le informazioni relative alla richiesta di approvvigionamento di un determinato prodotto, specificando: quantità di rifornimento, descrizione, nome del fornitore ed email del fornitore.		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
CodiceRichiesta	int	PRIMARY KEY	NOT NULL AUTO_INCREMENT
Fornitore	Varchar(50)		NOT NULL
EmailFornitore	Varchar(50)		NOT NULL
Descrizione	text		NOT NULL
QuantitàRifornimento	int		NOT NULL
Prodotto	int	FOREIGN KEY	NOT NULL

Nome Entità	Immagine_di_dettaglio		
Descrizione	Immagine che si focalizza su un particolare aspetto del prodotto venduto dal negozio		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli



CodiceImmagine	int	PRIMARY KEY	NOT NULL AUTO_INCREMENT
Prodotto	int	PRIMARY KEY FOREIGN KEY	NOT NULL
Contenuto	mediumblob		NOT NULL

Nome Entità	Ordine		
Descrizione	Le informazioni relative ad un ordine commissionato dal cliente al negozio da recapitare ad uno degli indirizzi di spedizione scelti dal cliente.		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
CodiceOrdine	int	PRIMARY KEY	NOT NULL AUTO_INCREMENT
Stato	enum(Richiesta effettuata, In lavorazione, Spedito, Preparazione incompleta)		NOT NULL
Email	Varchar(50)	FOREIGN KEY	NOT NULL
IndirizzoSpedizione	int		NOT NULL
TipoSpedizione	enum(Spedizione standard, Spedizione assicurata, Spedizione prime)		NOT NULL
TipoConsegna	enum(Domicilio, Punto di ritiro, Priority)		NOT NULL
DataOrdine	date		NOT NULL
OraOrdine	time		NOT NULL



<i>Nome Entità</i>	Composizione_Ordine		
<i>Descrizione</i>	Per ogni ordine di un cliente si tiene traccia dei prodotti acquistati, comprendenti le quantità richieste ed il prezzo del prodotto al momento dell'acquisto.		
<i>Nome campo</i>	<i>Tipo</i>	<i>Vincolo di chiave</i>	<i>Altri vincoli</i>
Ordine	int	PRIMARY KEY FOREIGN KEY	NOT NULL
Prodotto	int	PRIMARY KEY FOREIGN KEY	NOT NULL
QuantitàAcquistata	smallint		NOT NULL
PrezzoAcquistato	float		NOT NULL

<i>Nome Entità</i>	Report_di_spedizione		
<i>Descrizione</i>	Le informazioni relative ad un ordine spedito dal negozio dopo averlo elaborato: azienda logistica, imballaggio, data di spedizione e ora di spedizione.		
<i>Nome campo</i>	<i>Tipo</i>	<i>Vincolo di chiave</i>	<i>Altri vincoli</i>
NumeroReport	int	PRIMARY KEY	NOT NULL AUTO_INCREMENT
Corriere	Varchar(60)		NOT NULL
Imballaggio	Varchar(100)		NOT NULL
DataSpedizione	date		NOT NULL
OraSpedizione	time		NOT NULL
Ordine	int	FOREIGN KEY	NOT NULL

<i>Nome Entità</i>	Pagamento		
<i>Descrizione</i>	Le informazioni relative al pagamento di un ordine effettuato dal cliente: codice di pagamento, data di pagamento, ora di pagamento, importo totale.		
<i>Nome campo</i>	<i>Tipo</i>	<i>Vincolo di chiave</i>	<i>Altri vincoli</i>
CodicePagamento	int	PRIMARY KEY	NOT NULL AUTO_INCREMENT
Ordine	int	FOREIGN KEY	NOT NULL
DataPagamento	date		NOT NULL

OraPagamento	time		NOT NULL
Importo	float		NOT NULL

Nome Entità	Paypal		
Descrizione	Le informazioni relative al pagamento di un ordine effettuato dal cliente mediante Paypal.		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
CodicePagamento	int	PRIMARY KEY FOREIGN KEY	NOT NULL

Nome Entità	Contrassegno		
Descrizione	Le informazioni relative al pagamento di un ordine effettuato dal cliente mediante contrassegno.		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
CodicePagamento	int	PRIMARY KEY FOREIGN KEY	NOT NULL

Nome Entità	Carta_di_credito		
Descrizione	Le informazioni relative al pagamento di un ordine effettuato dal cliente mediante Paypal.		
Nome campo	Tipo	Vincolo di chiave	Altri vincoli
CodicePagamento	int	PRIMARY KEY FOREIGN KEY	NOT NULL
Titolare	Varchar(80)		NOT NULL
NumeroCarta	char(16)		NOT NULL

Per maggiori dettagli si consiglia la visione del documento di design del database presente nei documenti Deliverables rilasciati al cliente.

3.5 Controllo degli accessi e sicurezza

Si mostra la matrice degli accessi per poter tenere traccia delle operazioni che determinate categorie di utenti possono eseguire sugli oggetti del sistema software.

Attori Classi	GestioneOrdiniService	PagamentoService	GestioneCatalogoService	GestioneApprovvigionamentiService
Visitatore				
Cliente				
Utente (attore)	<ul style="list-style-type: none"> commissionaOrdine() 	<ul style="list-style-type: none"> effettuaPagamento() 		
Gestore degli ordini	<ul style="list-style-type: none"> visualizzaOrdiniEvasi() visualizzaOrdiniDaEvadere() preparazioneSpedizioneOrdine() 			<ul style="list-style-type: none"> visualizzaRichiesteFornitura() effettuaRichiestaApprovvigionamento()
Gestore del catalogo			<ul style="list-style-type: none"> visualizzaCatalogo() aggiuntaProdottoInCatalogo() rimozioneProdottoDaCatalogo() aggiornamentoSpecificheProdotto() aggiornamentoProdottoInVetrina() aggiornamentoPrezzoProdotto() aggiornamentoDisponibilit�Prodotto() inserimentoTopImmagine() inserimentoImmagineInGalleriaImmagine() cancellazioneImmagineInGalleria() 	

Attori Classi	RegistrazioneService	AutenticazioneService	GestioneWishlistService	GestioneCarrelloService	NavigazioneService
Visitatore	<ul style="list-style-type: none"> registraCliente() 			<ul style="list-style-type: none"> visualizzaCarrello() aggiungiAlCarrello() rimuoviDalCarrello() aumentaQuantitaNelCarrello() decrementaQuantitaNelCarrello() 	<ul style="list-style-type: none"> visualizzaProdotto() ricercaProdottoMenu() ricercaProdottoBar()
Cliente		<ul style="list-style-type: none"> login() resetPassword() 		<ul style="list-style-type: none"> visualizzaCarrello() aggiungiAlCarrello() rimuoviDalCarrello() aumentaQuantitaNelCarrello() decrementaQuantitaNelCarrello() 	<ul style="list-style-type: none"> visualizzaProdotto() ricercaProdottoMenu() ricercaProdottoBar()
Utente (attore)		<ul style="list-style-type: none"> login() resetPassword() aggiornaProfilo() aggiornaRubricaIndirizzi() 	<ul style="list-style-type: none"> recuperaWishlist() visualizzaWishlist() aggiungiProdottoInWishlist() rimuoviProdottoDaWishlist() 	<ul style="list-style-type: none"> visualizzaCarrello() aggiungiAlCarrello() rimuoviDalCarrello() aumentaQuantitaNelCarrello() decrementaQuantitaNelCarrello() svuotaCarrello() 	<ul style="list-style-type: none"> visualizzaProdotto() ricercaProdottoMenu() ricercaProdottoBar()
Gestore degli ordini		<ul style="list-style-type: none"> login() 			
Gestore del catalogo					



Il meccanismo di autenticazione degli attori al sistema è a singolo fattore: si richiederà l'inserimento dell'username e della password per poter accedere alle funzionalità riservate al gestore del catalogo, al gestore degli ordini ed all'utente. In particolare, la password verrà crittografata mediante un algoritmo di crittografia in modo da non memorizzarla in chiaro nella base di dati.

3.6 Controllo globale del software

Il flusso di controllo del sistema software proposto è event-driven.

Essendo il sistema interattivo, ogni funzionalità offerta dal sistema viene avviata in seguito ad un comando, rappresentato da un componente dell'interfaccia grafica, impartito dall'utente.

Pertanto, l'azione eseguita dall'utente scatenerà un evento, il quale verrà gestito da un apposito handler. L'handler, a sua volta, in base alle informazioni associate all'evento, indirizzerà il controllo del flusso di eventi al sottosistema che si occupa della logica di controllo.

Tale sottosistema, poi, si rivolge ai servizi per la logica applicativa.

3.7 Boundary conditions

In questo paragrafo verranno presentate i boundary condition use cases inerenti all'avvio del sistema, spegnimento del sistema, fallimento del sistema ed errore di accesso ai dati persistenti. I casi d'uso che descrivono le boundary conditions hanno come attore l'amministratore del sistema.

1. Avvio del sistema

Entry condition: L'amministratore è autenticato nel server.

Flusso di eventi:

1. L'amministratore esegue sulla macchina il comando che avvia il sistema.
2. Il sistema verifica la sanità dei dati persistenti e, se sani, rende le sue funzionalità disponibili agli utenti.

Exit condition: Il sistema viene avviato correttamente.

Flusso alternativo



Se nel punto 2 i dati persistenti risultano essere danneggiati, il sistema notifica tale problema all'amministratore di sistema e non effettua l'avvio.

2. Spegnimento del sistema

Entry condition: L'amministratore accede al server. Il sistema è stato precedentemente avviato e non è stato ancora spento.

Flusso di eventi:

1. L'amministratore invia un segnale di spegnimento al sistema.
2. Il sistema controlla se ci sono connessioni ancora attive da o verso l'esterno e, in assenza di tali connessioni, termina l'esecuzione.

Exit condition: Il sistema viene spento correttamente.

Flusso alternativo

Se nel punto 2 ci sono ancora connessioni attive, allora il sistema notifica tale situazione all'amministratore. A questo punto, il sistema attende una quantità di tempo per rispondere a eventuali richieste dall'esterno, non generando nuove connessioni se non allo scopo di rispondere a richieste già in corso.

Successivamente, il sistema controlla che non ci siano connessioni ancora aperte da o verso l'esterno e, se non ci sono, termina l'esecuzione del sistema.

Nel caso in cui continuino a permanere connessioni aperte, il sistema procederà con lo spegnimento forzato, recidendo tali connessioni.

3. Fallimento del sistema

Entry condition:

- Il sistema si arresta completamente;
- OR Il sistema non risponde alle richieste degli utenti;
- OR le prestazioni del sistema sono diminuite in modo significativo.

Flusso di eventi:

1. Il sistema comunica la situazione all'amministratore di sistema e termina.
2. L'amministratore esamina i log di sistema per identificare la causa del guasto.
3. L'amministratore di sistema esegue le opportune azioni per risolvere il guasto.
4. L'amministratore riavvia il sistema correttamente.

Exit condition: Il sistema è stato riavviato e funziona correttamente.



Flusso alternativo

Se nel punto 3 il guasto di sistema è causato da un problema hardware oppure l'amministratore non è in grado di identificare la causa del guasto, l'amministratore contatterà il supporto tecnico del fornitore del sistema.

4. Errore di accesso ai dati persistenti

Entry condition:

- il sistema non riesce ad accedere ai dati persistenti
- OR i dati persistenti risultano essere corrotti.

Flusso di eventi:

1. Il sistema comunica all'amministratore l'impossibilità di accedere ai dati persistenti.
2. Il sistema interrompe l'elaborazione di eventuali richieste da o verso l'esterno, mostrando un messaggio di errore agli utenti collegati.
3. L'amministratore esamina i log di sistema per identificare la causa del problema.
4. L'amministratore ripristina l'accessibilità o la sanità dei dati persistenti.
5. L'amministratore riavvia il sistema.

Exit condition: Il sistema riprende il normale funzionamento.

Flusso alternativo

Se nel punto 3 il guasto di sistema è causato da un problema hardware oppure l'amministratore non è in grado di identificare la causa del guasto, l'amministratore contatterà il supporto tecnico del fornitore del sistema.



3.8 Servizi dei sottosistemi

In questa sezione si descrivono i servizi di ogni sottosistema precedentemente elencati.

Si precisa che in *RegistrazioneService* ci sono i servizi *registraGestoreOrdini* e *registraGestoreCatalogo* per effettuare la registrazione, rispettivamente, del gestore degli ordini e del gestore del catalogo.

Vista l'importanza del ruolo ricoperto da tale tipologia di utenti all'interno del negozio e per motivi di sicurezza, il nostro sistema gestisce esclusivamente la registrazione dei clienti.

La registrazione dei suddetti dipendenti verrà gestita direttamente dall'amministratore di sistema per garantire la protezione dei dati e il rispetto delle normative vigenti.

Servizio	Descrizione	Interfaccia	Sottosistema
registraCliente	Permette di registrare un cliente al sistema.	RegistrazioneService	Registrazione
login	Permette di autenticare un cliente o un dipendente (fino adesso gestore catalogo e gestore degli ordini) al sistema mediante l'inserimento di username e password.	AutenticazioneService	Autenticazione
resetPassword	Effettua la reimpostazione della password di un cliente registrato al sistema. In questo modo viene impostata una nuova password.	AutenticazioneService	Autenticazione
aggiornaProfilo	Consente di aggiornare il numero di telefono o l'email dell'utente.	AutenticazioneService	Autenticazione
aggiornaRubricaIndirizzi	Tale servizio permette di	AutenticazioneService	Autenticazione

	aggiornare la rubrica di indirizzi di spedizione di un utente: aggiungere/rimuovere/aggiornare un indirizzo di spedizione.		
visualizzaCarrello	Recupera i prodotti inseriti nel carrello dal cliente e li visualizza.	GestioneCarrelloService	GestioneCarrello
aggiungiAlCarrello	Il servizio aggiunge un prodotto nel carrello virtuale. Qualora il prodotto fosse presente già nel carrello, verrà incrementata la quantità di quel prodotto nel carrello.	GestioneCarrelloService	GestioneCarrello
rimuoviDalCarrello	Il servizio rimuove un prodotto selezionato dal carrello.	GestioneCarrelloService	GestioneCarrello
aumentaQuantitaNelCarrello	Il servizio aumenta la quantità di un prodotto nel carrello.	GestioneCarrelloService	GestioneCarrello
decrementaQuantitaNelCarrello	Diminuisce la quantità di un prodotto nel carrello.	GestioneCarrelloService	GestioneCarrello
svuotaCarrello	Rimuove i prodotti presenti nel carrello (es. a seguito di un acquisto).	GestioneCarrelloService	GestioneCarrello
recuperaWishlist	Il servizio recupera la lista dei desideri di un cliente per codice (il sistema consentirà di generare una sola wishlist per utente).	GestioneWishlistService	GestioneWishlist

visualizzaWishlist	Il servizio consente di visualizzare i prodotti presenti nella wishlist di un cliente.	GestioneWishlistService	GestioneWishlist
aggiungiProdottoInWishlist	Il servizio inserisce un prodotto nella wishlist. Se il prodotto è già presente, allora non verrà effettuato alcun inserimento.	GestioneWishlistService	GestioneWishlist
rimuoviProdottoDaWishlist	Il servizio permette di rimuovere un prodotto, selezionato dal cliente, dalla wishlist.	GestioneWishlistService	GestioneWishlist
ricercaProdottoMenu	Il servizio consente di eseguire la ricerca dei prodotti per selezione della categoria dal menubar.	NavigazioneService	Navigazione
ricercaProdottoBar	Il servizio permette di eseguire la ricerca di un prodotto mediante la barra di ricerca.	NavigazioneService	Navigazione
visualizzaProdotto	Permette di visualizzare le specifiche complete di un prodotto.	NavigazioneService	Navigazione
commissionaOrdine	Il servizio effettua la creazione di un ordine contenente i prodotti del carrello (check-out del carrello).	GestioneOrdiniService	GestioneOrdini
visualizzaOrdiniEvasi	Fornisce gli ordini evasi dal negozio online.	GestioneOrdiniService	GestioneOrdini
visualizzaOrdiniDaEvadere	Visualizza gli ordini commissionati al negozio online.	GestioneOrdiniService	GestioneOrdini



preparazioneSpedizioneOrdine	Il servizio permette di elaborare un ordine commissionato al negozio affinché possa essere spedito al cliente.	GestioneOrdiniService	GestioneOrdini
effettuaPagamento	Il servizio effettua il pagamento dell'ordine del cliente (fino adesso con le seguenti modalità: carta di credito, Paypal, contrassegno).	PagamentoService	Pagamento
visualizzaRichiesteFornitura	Fornisce le richieste di approvvigionamento effettuate.	GestioneApprovvigionamentiService	GestioneApprovvigionamenti
effettuaRichiestaApprovvigionamento	Il servizio permette di creare una richiesta di approvvigionamento di un prodotto.	GestioneApprovvigionamentiService	GestioneApprovvigionamenti
visualizzaCatalogo	Si visualizza il catalogo dei prodotti venduti dal negozio online.	GestioneCatalogoService	GestioneCatalogo
aggiuntaProdottoInCatalogo	Il servizio permette di aggiungere un prodotto nel catalogo.	GestioneCatalogoService	GestioneCatalogo
rimozioneProdottoDaCatalogo	Il servizio permette di cancellare un prodotto dal catalogo del negozio.	GestioneCatalogoService	GestioneCatalogo
aggiornamentoSpecificheProdotto	Il metodo permette di aggiornare le seguenti specifiche del prodotto: modello, marca, descrizione in evidenza, descrizione	GestioneCatalogoService	GestioneCatalogo

	dettagliata, categoria, sottocategoria.		
aggiornamentoProdottoInVetrina	Consente di inserire/rimuovere un prodotto da una vetrina virtuale del negozio per pubblicità dei prodotti del negozio.	GestioneCatalogoService	GestioneCatalogo
aggiornamentoDisponibilitàProdotto	Il servizio permette di aggiornare la quantità di scorte in magazzino del prodotto.	GestioneCatalogoService	GestioneCatalogo
aggiornamentoPrezzoProdotto	Il servizio permette di aggiornare il prezzo di un prodotto.	GestioneCatalogoService	GestioneCatalogo
inserimentoTopImmagine	Il servizio permette di inserire l'immagine di presentazione di un prodotto.	GestioneCatalogoService	GestioneCatalogo
inserimentoImmagineInGalleriaImmagini	Permette di aggiungere un'immagine di dettaglio di un prodotto.	GestioneCatalogoService	GestioneCatalogo
cancellazioneImmagineInGalleria	Consente di rimuovere un'immagine di dettaglio di un prodotto dalla galleria di immagini ad esso associate.	GestioneCatalogoService	GestioneCatalogo



4. Glossario

In questa sezione sono raccolte le definizioni di alcuni termini utilizzati nel documento.

Termine	Definizione
COTS (Commercial Off-The-Shelf)	Si riferisce a software o componenti software che sono disponibili per l'acquisto da terze parti anziché essere sviluppati internamente.
Piattaforma	Base software o hardware su cui sono sviluppate o eseguite applicazioni.
Algoritmo crittografico	Un algoritmo crittografico si basa su una funzione matematica attraverso cui si nasconde il testo, rendendolo di fatto cifrato. Dopo il processo di crittografia, poi, l'accesso alle informazioni criptate avviene attraverso una chiave digitale, capace di rendere accessibili i dati.
Messa in evidenza	Ogni sito e-commerce ha la possibilità di creare delle vetrine virtuali in cui pubblicizzare i prodotti venduti. La messa in evidenza di un prodotto specifica se il prodotto è presente in almeno una vetrina virtuale dell'e-commerce.