



UNIVERSITÀ DEGLI STUDI DI SALERNO  
FACOLTÀ DI SCIENZE MM.FF.NN.  
CORSO DI LAUREA TRIENNALE IN INFORMATICA  
CORSO DI INGEGNERIA DEL SOFTWARE

# Gamerland

## **System Design Document (S.D.D)**

**VERSIONE 1.2**

**Anno Accademico 2017/18**

## Top Manager:

Professore
Prof. De Lucia Andrea
Prof. Francese Rita

## Partecipanti:

Nome	Matricola
Capone Natale	0512103586
Stanzione Roberto	0512103763
Iuliano Gianpaolo	0512103514

## Revision History:

Data	Versione	Descrizione	Autore
17/12/2017	1.0	Inizio stesura documento	Tutto il team
18/12/2017	1.1	Aggiunta matrice accessi, boundary condition e diagrammi (deployment e component)	Tutto il team
05/02/2018	1.2	Finalizzazione documento	Tutto il team

# Sommario

- 1. Introduzione ..... 4
  - 1.1 obiettivi del sistema ..... 4
  - 1.2 Design goals ..... 6
  - 1.3 Glossario ..... 9
- 2. Architettura del sistema proposto ..... 9
  - 2.1 Panoramica ..... 9
  - 2.2 Decomposizione del sistema .....10
  - 2.3 Mapping Hardware/Software .....13
  - 2.4 Controllo degli accessi e sicurezza .....14
  - 2.5 Boundary condition .....15
  - 2.6 servizi dei sottosistemi .....16

## 1. Introduzione

**Gamerland** è un sito che nasce per soddisfare le esigenze di quegli utenti che vogliono acquistare prodotti relativi al settore dei videogiochi senza dover lasciare le proprie abitazioni.

**Gamerland** è un software Web-based per la gestione di uno store online che permette di apportare miglioramenti sulla gestione di prodotti e-commerce, basati soprattutto sulla vendita di videogames, console ed accessori.

**Gamerland** nasce con l'idea di differenziarsi rispetto alla concorrenza mostrando all'utente un'interfaccia semplice e dal design pulito. L'elemento cardine che contraddistingue la nostra piattaforma dai siti rivali, quindi, è la semplicità di utilizzo e la centralità del prodotto, senza troppi fronzoli.

**Gamerland** si propone di offrire notevoli vantaggi per gli utenti, volti all'efficienza, sotto forma di servizi. Infatti, l'utente ha la possibilità di poter gestire l'acquisto dei prodotti desiderati in modo pratico e veloce, semplicemente effettuando una registrazione al sito. Nella registrazione vengono richiesti i dati anagrafici del cliente, per l'identificazione dell'ordine, e l'indirizzo di spedizione.

### 1.1 obiettivi del sistema

- La funzione principale che il software andrà a svolgere dal punto di vista utente sarà quella di acquistare prodotti presenti nel catalogo, diviso per categorie, effettuando:
  - la registrazione, operazione comunemente utilizzata nei software dedicati alla gestione di store online;
  - la visualizzazione di ogni prodotto, presente nelle relative sezioni dello store, con annessa descrizione, disponibilità e prezzo;
  - l'aggiunta dei prodotti desiderati in un carrello virtuale;
  - La conferma di acquisto dei prodotti presenti nel carrello.
- La funzione principale del software dal punto di vista dell'amministratore, sarà quella di aggiungere o rimuovere prodotti dal catalogo tenendo sempre aggiornato lo store online. L'amministratore quindi andrà a

svolgere come attore due principali operazioni, che sono l'aggiunta e la cancellazione dei prodotti dal catalogo dello store.

- L'aggiunta dei nuovi prodotti avviene con l'uso di appositi form di inserimento, che riguarderanno principalmente il nome del prodotto, la piattaforma, il prezzo, la quantità e la descrizione.
  - La rimozione dei prodotti avviene semplicemente navigando nel catalogo, in modalità admin, e cliccando sull'apposita voce di rimozione presente nel menù.
- 
- La funzione principale del software dal punto di vista del gestore magazzino, sarà quella di visualizzare e provvedere alla spedizione degli ordini effettuati dai clienti.
    - Il gestore magazzino visualizza gli ordini spediti e da spedire;
    - Nel caso in cui un ordine è nello stato da spedire, il gestore provvederà alla spedizione andando a modificare lo stato dell'ordine.
- 
- La funzione principale del software dal punto di vista del proprietario dell'attività, sarà quella di andare a visualizzare i ricavi dello store online.
    - Nello specifico il proprietario andrà a visionare il bilancio dello store.

## 1.2 Design goals

I design goal identificati per il software **Gamerland** sono i seguenti:

### **Criteri di performance**

- **Tempo di risposta:**

Essendo un sito di e-commerce, il software verrà utilizzato solo tramite Internet e di conseguenza, la risposta del software all'interazione dell'utente, sarà condizionata principalmente dalla rete.

Per la visualizzazione delle pagine il tempo di risposta massimo è di 1 secondo.

Per l'aggiunta o rimozione di prodotti al carrello il tempo di risposta è circa di 0,5 secondi.

Per la conferma di acquisto il tempo di risposta del sistema è di circa 0,5 secondi.

- **Memoria:**

La dimensione complessiva del sistema dipende dalla memoria utilizzata per il mantenimento del database.

### **Criteri di affidabilità**

- **Robustezza:**

Eventuali input non validi immessi dall'utente saranno opportunamente segnalati attraverso messaggi di errore.

- **Affidabilità:**

I prodotti e le varie informazioni visualizzate sono coerenti con i dati presenti nel database.

- **Disponibilità:**

Una volta realizzato il sistema, e dopo la messa online, esso sarà disponibile agli utenti in qualsiasi momento.

- **Tolleranza ai guasti:**

Il sistema può subire guasti dovuti al sovraccarico del database con successivo fallimento. Per ovviare al problema, periodicamente è previsto un salvataggio dei dati sotto forma di codice SQL necessario per la rigenerazione del database.

- **Security:**

L'accesso al sistema è garantito mediante una username e una password.

La sicurezza del database è garantita dal fatto che lo stesso è in locale, quindi accessibile solo all'addetto ai lavori.

## **Criteri di costo**

- **Costi di sviluppo:**

È stimato un costo complessivo di 540 ore per la progettazione e lo sviluppo del sistema (180 ore per ogni team member).

## **Criteri di manutenzione**

- **Estendibilità:**

Il sistema è estendibile in quanto esso può essere esteso con nuove funzionalità nel corso del tempo.

- **Adattabilità:**

Il sistema può essere modificato, e quindi adattato, per realizzare una qualsiasi piattaforma di e-commerce.

- **Tracciabilità dei requisiti:**

La tracciabilità dei requisiti è possibile grazie ad una matrice di tracciabilità, attraverso la quale è possibile retrocedere al requisito associato ad ogni parte del progetto.

La tracciabilità è garantita dalla fase di progettazione fino al testing.

- **Portabilità:**

Il sistema sarà portabile in quanto l'interazione avviene mediante un browser senza interazione con il sistema sottostante, c'è quindi indipendenza dal sistema operativo.

## **Criteri utenti finali**

- **Usabilità:**

L'usabilità di un sistema può essere analizzata considerando diverse caratteristiche. Questo sistema sarà molto semplice da apprendere senza la consultazione di documentazione associata.

L'intuitività è garantita in quanto il sistema avrà una buona prevedibilità, cioè la risposta del sistema ad un'azione utente sarà corrispondente alle aspettative.

- **Utilità:**

La navigazione dell'utente verrà supportata nel miglior modo possibile dal sistema. L'utente verrà guidato passo dopo passo al fine di rendere qualsiasi operazione il più semplice e intuitiva possibile. Tutto ciò permette all'utente di effettuare acquisti comodamente da qualsiasi luogo in cui si trova con semplicità, risparmiando la solita trafila del centro commerciale.



## 1.3 Glossario

<b>RAD</b>	Requirements Analysis Document.
<b>SDD</b>	System Design Document.
<b>ODD</b>	Object Design Document.
<b>DB</b>	Database.
<b>MVC</b>	Model Control View
<b>EVENT DRIVEN</b>	Tipologia di controllo del flusso, il controllo risiede in un dispatcher che nel nostro caso è l'utente.

## 2. Architettura del sistema proposto

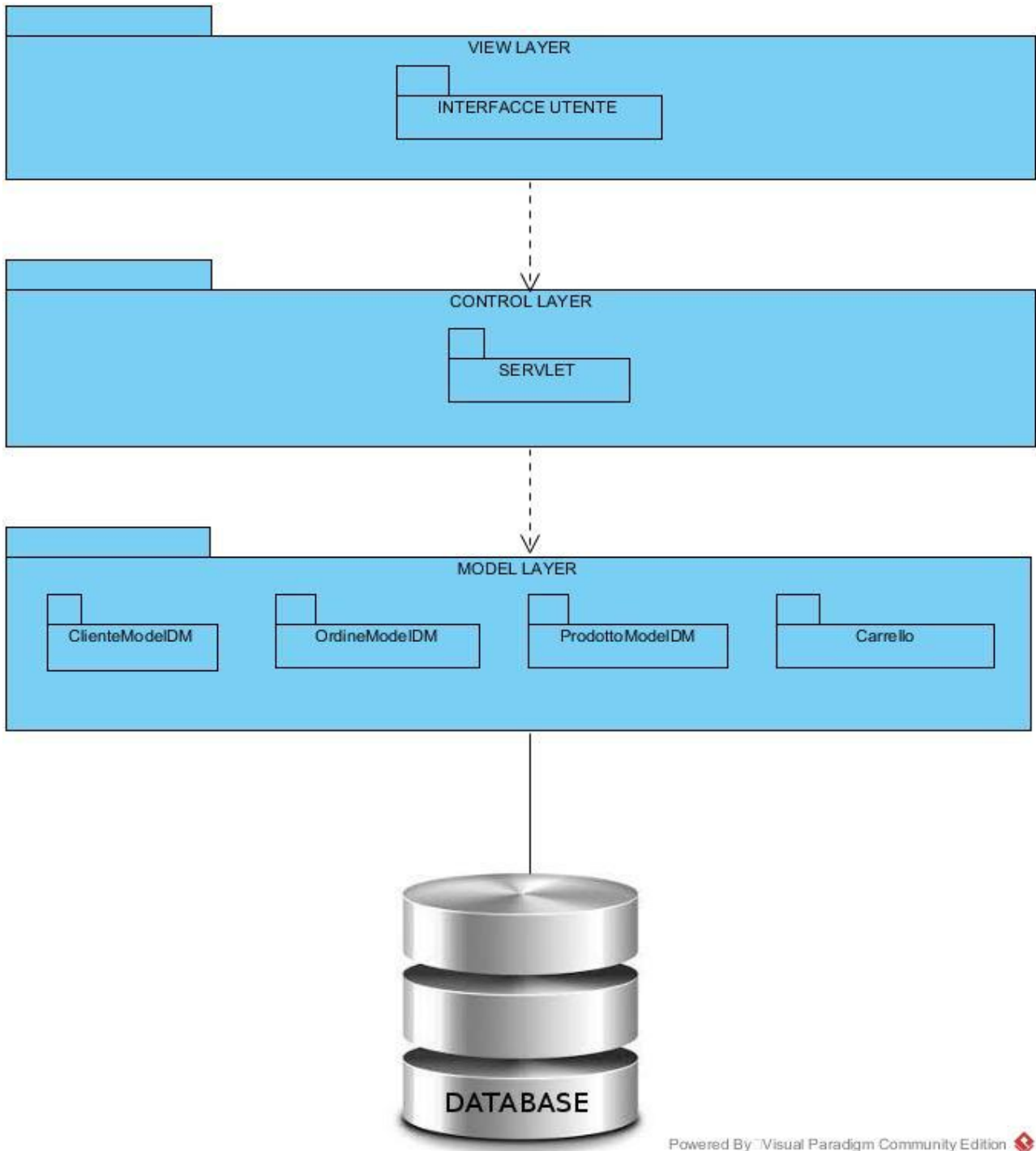
### 2.1 Panoramica

Il sistema da noi proposto è un'applicazione web, in locale per motivi di sicurezza. L'obiettivo che si pone è quello di realizzare un sito di E-commerce dedicato alla vendita di videogiochi, console e accessori. L'acquisto dei prodotti citati avviene tramite l'aggiunta al carrello con semplici operazioni di selezione nei vari cataloghi, presenti nel sistema, e una conferma di acquisto. L'utente per effettuare l'acquisto deve essere registrato al sistema, per effettuare la registrazione l'utente non deve far altro che compilare degli appositi form i cui dati verranno poi salvati in un database relazionale. Per tutelare la sicurezza dell'utente l'autenticazione avviene tramite username e password personali. In un'apposita sezione del sistema è presente un'area riservata per ciascun utente, dove è possibile modificare password e indirizzo di spedizione, o cancellare il proprio account. Dal punto di vista della gestione dei cataloghi è presente un'apposita figura, l'amministratore, che accedendo con le proprie credenziali al sistema va a modificare i prodotti nei cataloghi. Le altre figure che potranno effettuare l'accesso al sistema sono: il proprietario dell'attività a cui è permesso visualizzare i ricavi dell'attività e il gestore magazzino che tramite le sue credenziali, accedendo ad un'apposita sezione, va a modificare lo stato degli ordini effettuati dagli utenti.

Il sistema verrà realizzato tramite il modello MVC, ovvero composto da elementi Model che rappresentano gli oggetti persistenti del database, Control che implementano la logica di Business ed infine i View che si occupano della visualizzazione dell'interfaccia grafica.

## 2.2 Decomposizione del sistema

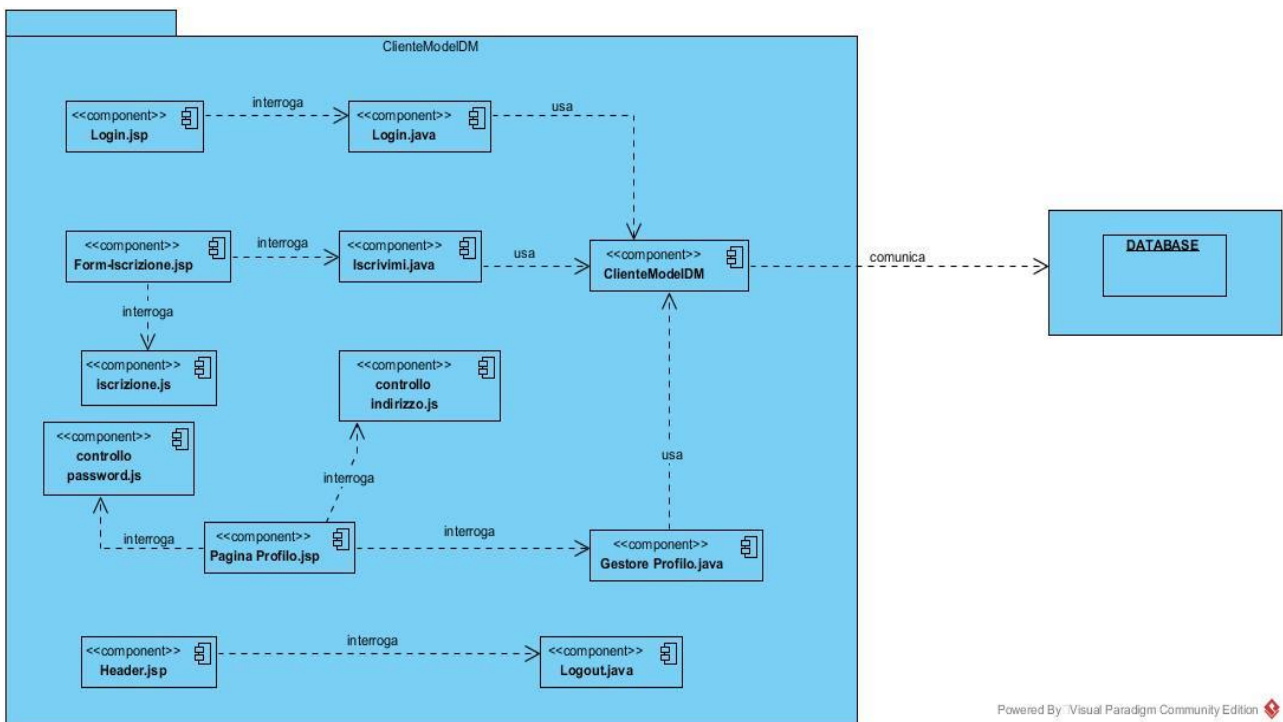
Il sistema proposto è composto da diversi sottosistemi, ognuno dei quali offre determinati servizi, questi sottosistemi sono: ClienteModelDM, OrdineModelDM, ProdottoModelDM e Carrello. Il sistema è strutturato secondo un architettura MVC. Di seguito viene mostrato il diagramma della decomposizione:



Ognuno di questi sottosistemi è composto da alcune componenti (pagine jsp, servlet, manager dei dati).

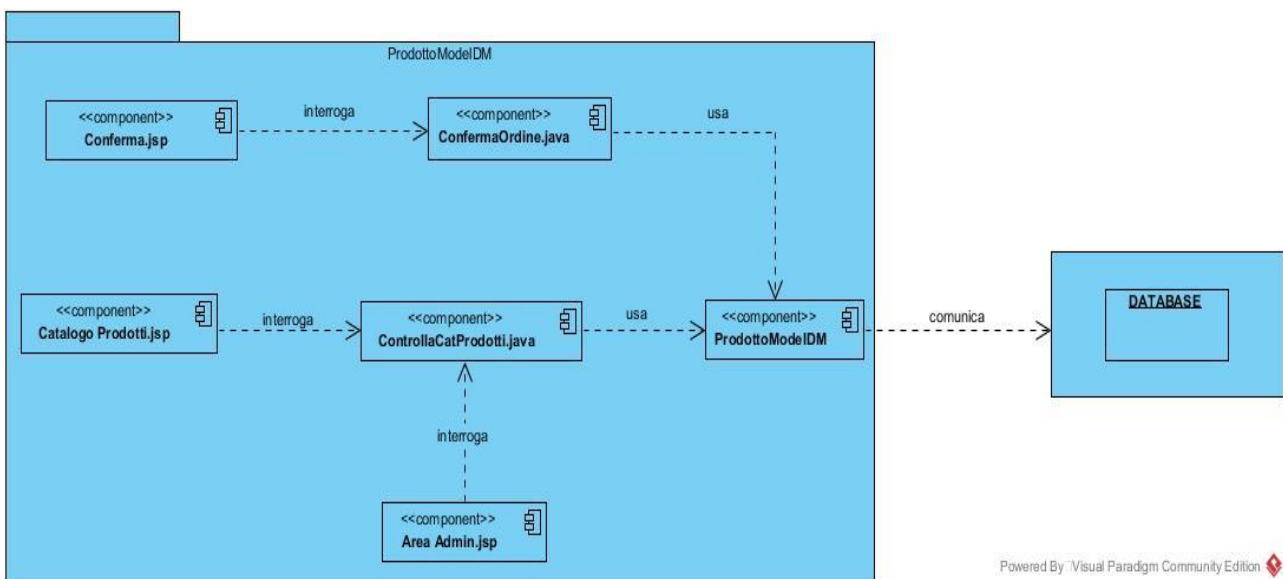
Di seguito vengono mostrati i component diagram per ogni sottosistema.

## ClienteModelDM



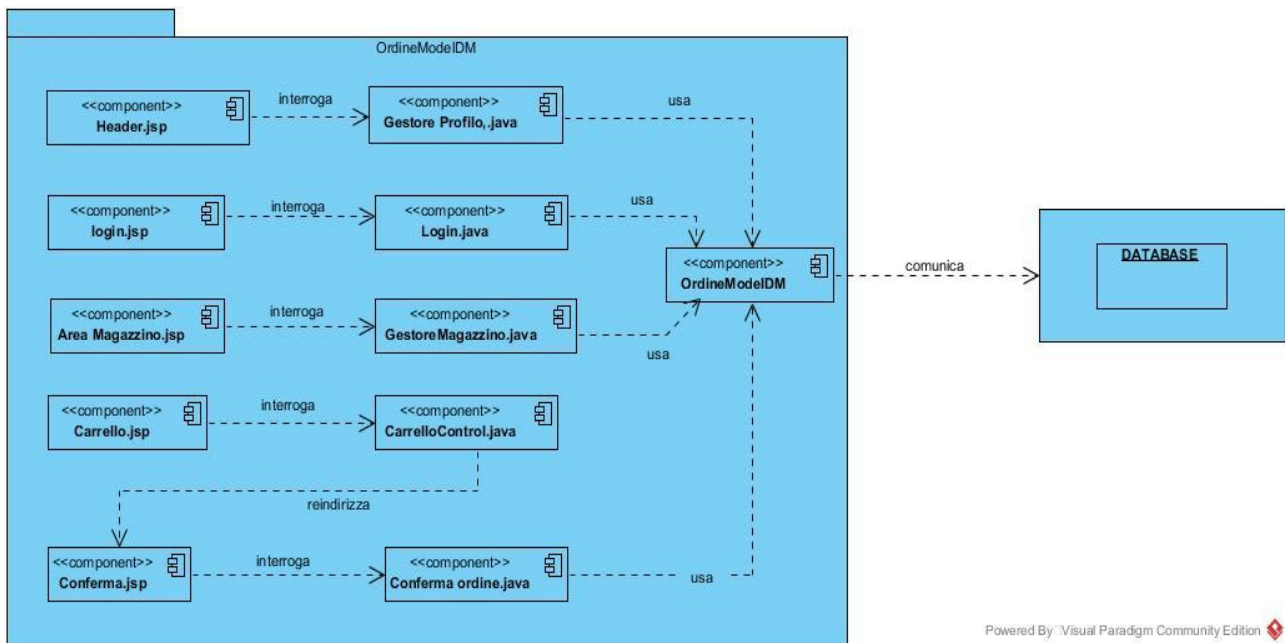
Powered By Visual Paradigm Community Edition

## ProdottoModelDM

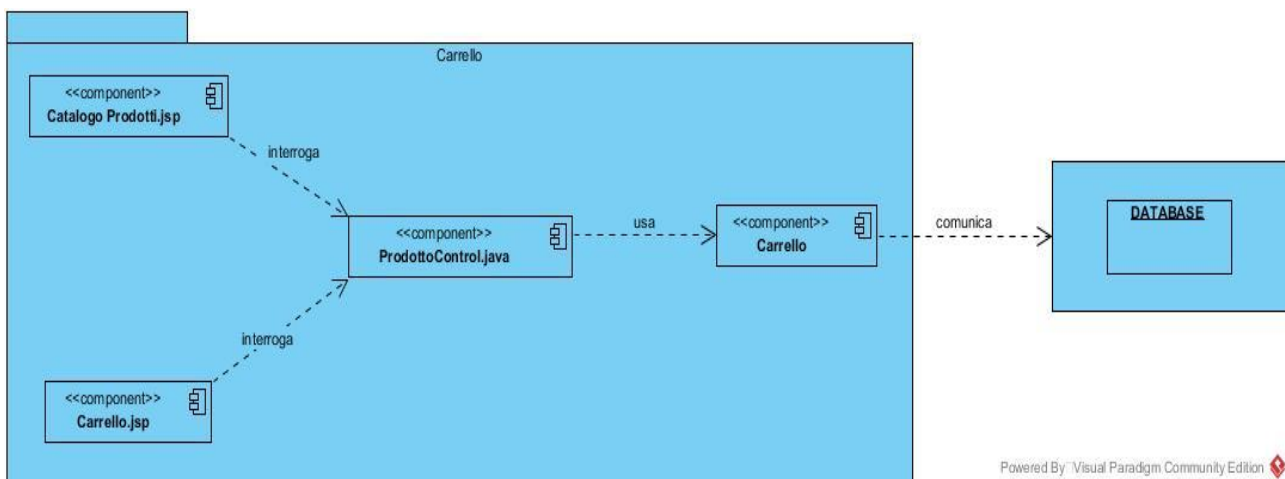


Powered By Visual Paradigm Community Edition

## OrdineModelDM



## Carrello



## 2.3 Mapping Hardware/Software

### Web Server

Il server utilizzato è Apache versione 8.0

### View layer

L'utente utilizza il sistema mediante un Browser installato all'interno del suo calcolatore (ad es. Opera, Firefox, Chrome). L'interfaccia del sistema è realizzata tramite delle pagine JSP.

### Control layer

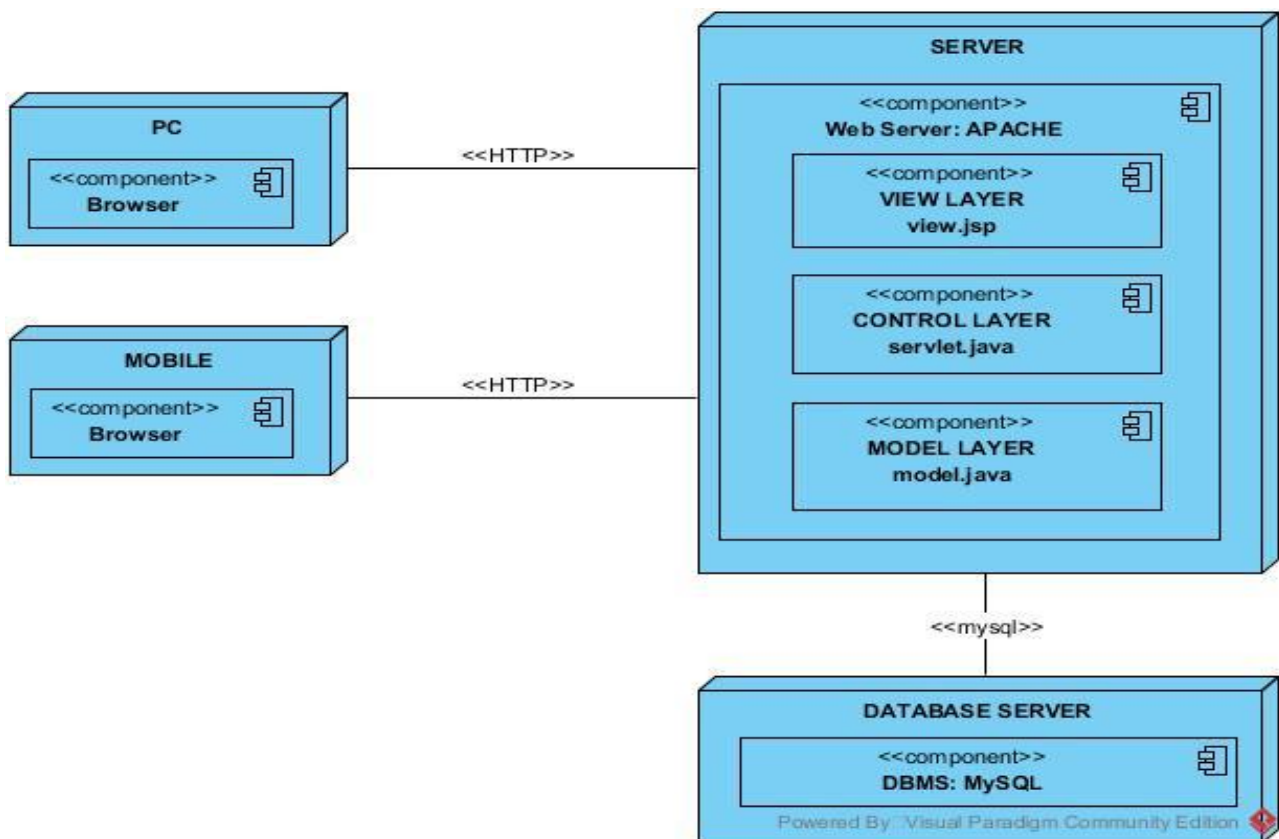
Il sistema, e quindi le funzionalità, sono implementate in linguaggio JAVA. Il Sistema è strutturato in Servlet che si occupano di creare una comunicazione tra il Model layer e il View layer. Il codice html viene inviato al browser del client.

### Model layer

Rappresenta il collegamento con il server da parte del sistema e si occupa di tutte le richieste di accesso e modifiche sui dati permanenti presenti nel database.

### Database Server

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



## 2.4 Controllo degli accessi e sicurezza

Gamerland è un sistema multiutente, quindi attori differenti hanno il permesso di eseguire diverse operazioni su vari insiemi di oggetti. Per schematizzare al meglio il controllo degli accessi abbiamo suddiviso per tipologia di utente le azioni consentite, al fine di ottenere una visione più compatta e dettagliata grazie ad una matrice degli accessi riportata di seguito:

### Matrice degli accessi

ATTORE/SOTTOSISTEMA	ClienteModelDM	ProdottoModelDM	OrdineModelDM	Carrello
UTENTE	Login, logout, iscrizione, cambia indirizzo e città, cambia password, elimina account		Acquisto prodotti, visualizza ordini effettuati	Aggiungi al carrello, rimuovi dal carrello
AMMINISTRATORE	Login, logout, Cambia password	Aggiungi al catalogo, rimuovi dal catalogo		
GESTORE MAGAZZINO	Login, logout, cambia password		Modifica stato dell'ordine	
PROPRIETARIO ATTIVITA'	Login, logout, cambia password		Visualizza ricavi	

## 2.5 Boundary condition

Il sistema si avvierà con il caricamento del database e di tutti i dati contenuti al suo interno. Il sistema risulta pronto all'uso e disponibile all'invio dei dati in esso. L'User Interface carica gli script necessari alla composizione ed alla visualizzazione della homepage. Dalla barra navigazionale presente nella home, un utilizzatore può effettuare il login per poi interagire con il sistema. Ogni singolo sottosistema non viene terminato ma, resta latente se non direttamente invocato dal sistema. Se un sottosistema risulta momentaneamente non disponibile, allora il resto del sistema ne verrà informato attraverso un cambio comportamentale del sistema, dovuto alla gestione delle eccezioni. Di conseguenza, il comportamento conseguente ad un malfunzionamento, non causerà un riavvio di sistema ma si adatterà al tipo di failure in cui si imbatte. Le condizioni limite riguardano l'accensione e lo spegnimento del sistema per quanto riguarda il lato Server. Dal lato Client si riferiscono agli errori di connessione al server.

### StartServer

**Attore:** amministratore

**Entry condition:** L'amministratore ha effettuato il login sul server

**Flusso di eventi:** -

L' amministratore esegue il comando di startup del sistema Il sistema esegue le operazioni di inizializzazione, andando a leggere dal database i dati necessari alle normali funzionalità

**Exit condition:** Il sistema è inizializzato correttamente ed è disponibile

### ShutdownServer

**Attore:** amministratore

**Entry condition:** L'amministratore ha effettuato il login sul server

**Flusso di eventi:**

L' amministratore esegue il comando di shutdown del sistema.

Il sistema esegue le operazioni di terminazione, rilasciando tutte le risorse utilizzate e salvando i dati rimanenti sul database per recuperarli in seguito.

**Exit condition:** Il sistema è spento correttamente

## 2.6 servizi dei sottosistemi

- **ClienteModelDM**

<b>Sottosistema</b>	ClienteModelDM
<b>Descrizione</b>	Sottosistema che gestisce la registrazione di un utente, l'autenticazione di tutte e quattro le tipologie di utilizzatori del sistema, le modifiche alla password, all'indirizzo e alla città di un account e la cancellazione di un account.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
Iscrizione	Permette ai nuovi utenti del sistema di registrarsi nel sistema.
Login	Permette ad un utilizzatore di poter effettuare l'accesso al sistema.
Logout	Permette ad un utilizzatore di uscire dal sistema.
Cambia indirizzo e città	Permette di modificare l'indirizzo e la città di spedizione inseriti durante l'iscrizione.
Modifica password	Permette di modificare la password inserita durante l'iscrizione.
Elimina account	Permette di eliminare l'account dell'utente dal sistema.



- **ProdottoModelDM**

<b>Sottosistema</b>	ProdottoModelDM
<b>Descrizione</b>	Sottosistema che gestisce l'aggiunta e la rimozione dei prodotti nei vari cataloghi
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
Aggiungi al catalogo	Permette all'amministratore del sistema di aggiungere nuovi prodotti ai cataloghi
Rimuovi dal catalogo	Permette all'amministratore del sistema di rimuovere un prodotto dai cataloghi

- **OrdineModelDM**

<b>Sottosistema</b>	OrdineModelDM
<b>Descrizione</b>	Sottosistema che gestisce l'acquisto dei prodotti e la visualizzazione degli ordini effettuati parte di un utente. Inoltre, gestisce la modifica dello stato dell'ordine da parte del gestore del magazzino e la visualizzazione dei ricavi da parte del proprietario dell'attività.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
Acquisto prodotti	Permette, ad un utente registrato nel sistema, di effettuare l'acquisto dei prodotti contenuti nel carrello.
Visualizza ordini effettuati	Permette, ad un utente registrato nel sistema, di visualizzare gli ordini effettuati.
Modifica stato dell'ordine	Permette, al gestore del magazzino, di modificare lo stato di un ordine.
Visualizza ricavi	Permette, al proprietario dell'attività, di visualizzare i ricavi provenienti da tutti gli ordini effettuati nel sistema.

- Carrello

<b>Sottosistema</b>	Carrello
<b>Descrizione</b>	Sottosistema che gestisce l'aggiunta e la rimozione di un prodotto dall'istanza del carrello nel database.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
Aggiunta al carrello	Permette, ad un utente registrato nel sistema, di aggiungere un prodotto nel carrello aggiornando il database
Rimuovi dal carrello	Permette, ad un utente registrato nel sistema, di rimuovere un prodotto dal carrello aggiornando il database