



Unisa Cinema

UNIVERSITA' DEGLI STUDI DI SALERNO  
FACOLTA' DI SCIENZE MM. FF. NN.  
CORSO DI LAUREA TRIENNALE IN INFORMATICA  
CORSO DI INGEGNERIA DEL SOFTWARE



# UNISA CINEMA

## SDD – System Design Document

Anno Accademico 2017/18

Top Manager:

Professori
Prof. De Lucia Andrea
Prof. Francese Rita

Partecipanti:

Nome	Matricola
Amato Federica	0512103606
Cosenza Giuseppe	0512103486
De Sio Maria Grazia	0512103594
Pizzo Domenico	0512103652

Revision History:

Data	Versione	Descrizione	Autore
10/02/2018	1.0	Prima stesura del documento	Team members
04/01/2018	2.0	Stesura del documento	Domenico Pizzo Federica Amato Maria Grazia De Sio
02/02/2018	3.0	Revisione del documento	Domenico Pizzo Federica Amato Maria Grazia De Sio
25/02/2018	3.1	Ultima revisione del documento	Domenico Pizzo Federica Amato Maria Grazia De Sio Giuseppe Cosenza

## Sommario

<b>1.Introduzione.....</b>	<b>4</b>
<b>1.1 Scopo del sistema .....</b>	<b>4</b>
<b>1.2 Obiettivi di design .....</b>	<b>4</b>
<b>1.2.1 Criteri di Performance .....</b>	<b>4</b>
<b>1.2.2 Criteri di Affidabilità .....</b>	<b>5</b>
<b>1.2.3 Criteri di Manutenzione.....</b>	<b>6</b>
<b>1.2.4 Criteri per l'Utente Finale .....</b>	<b>6</b>
<b>1.3 Definizioni, acronimi e abbreviazioni.....</b>	<b>6</b>
<b>1.4 Riferimenti .....</b>	<b>7</b>
<b>1.5 Panoramica .....</b>	<b>7</b>
<b>2. Architettura del Software Corrente .....</b>	<b>7</b>
<b>3. Architettura del Software Proposto.....</b>	<b>7</b>
<b>3.1 Panoramica .....</b>	<b>7</b>
<b>3.2 Decomposizione del sistema .....</b>	<b>9</b>
<b>3.2.1 Utente non registrato .....</b>	<b>10</b>
<b>3.2.2 Cliente .....</b>	<b>11</b>
<b>3.2.3 Amministratore.....</b>	<b>12</b>
<b>3.2.4 Operatore.....</b>	<b>13</b>
<b>3.2.5 Diagramma Generale .....</b>	<b>14</b>
<b>3.3 Mapping Hardware/Software .....</b>	<b>15</b>
<b>3.4 Gestione dei dati persistenti .....</b>	<b>19</b>
<b>3.5 Controllo degli accessi e della sicurezza.....</b>	<b>19</b>
<b>3.6 Controllo del software globale .....</b>	<b>19</b>
<b>3.7 Condizione boundary.....</b>	<b>20</b>
<b>3.7.1 Avvio del sistema .....</b>	<b>20</b>
<b>3.7.2 Terminazione del sistema.....</b>	<b>20</b>
<b>3.7.3 Fallimento del sistema.....</b>	<b>20</b>
<b>4. Servizi dei sottosistemi .....</b>	<b>24</b>
<b>4.1 Gestione Account.....</b>	<b>24</b>
<b>4.2 Gestione Film .....</b>	<b>24</b>
<b>4.3 Gestione Spettacolo.....</b>	<b>25</b>
<b>4.4 Gestione Acquisto .....</b>	<b>25</b>
<b>4.5 Gestione Promozione .....</b>	<b>25</b>
<b>5.GLOSSARIO.....</b>	<b>26</b>

# 1.Introduzione

## 1.1 Scopo del sistema

La gestione di un canale di vendita online risulta un lavoro molto oneroso sia in termini di tempo che di manutenibilità. Chi opera su applicazioni dove si debbono apportare in modo frequente operazioni di modifica periodiche, spesso si trova a dover effettuare un lavoro ripetitivo e dispendioso in termini di tempo.

Altre difficoltà, non del tutto banali, riguardano soprattutto la poca competenza che alcuni utenti possono avere nell'utilizzo di portali Web, ciò implica necessariamente l'utilizzo di interfacce intuitive e di facile comprensione ossia user friendly.

Il sistema Unisa Cinema si pone lo scopo di facilitare la gestione e le modifiche del proprio sistema in modo agevole all'amministratore di tale software, così da permettere lo svolgimento di qualsiasi tipo di operazione qual si voglia effettuare. Principalmente si ha l'obiettivo di realizzare un'interfaccia intuitiva e semplice, che abbia lo scopo di poter far utilizzare questa piattaforma anche ad utenti che hanno poca dimestichezza con il mondo del Web.

## 1.2 Obiettivi di design

Il sistema Unisa Cinema deve poter essere il più efficiente ed intuitivo possibile. Tale efficienza sarà costruita attraverso rapidi tempi di risposta ad ogni genere di input ma anche differenti politiche di tolleranza all'errore. In più si punterà ad aver una buona manutenibilità attraverso inserimento di nuove funzionalità. Sarà intuitivo attraverso agevoli interfacce. Tutto ciò ci porta a considerare quattro distinte classi: Performance, Affidabilità, Manutenzione ed Utente Finale.

### 1.2.1 Criteri di Performance

<b>Tempo di Riposta</b>	Unisa Cinema deve essere reattivo per tutte le operazioni più immediate come l'aggiunta di un film. Per operazioni come il caricamento dei film in programmazione, deve garantire dei tempi di risposta nell'ordine delle decine di secondi, ma, essendo un sistema web, molto dipenderà dalla qualità della connessione e dalla congestione della rete sul sistema online.
<b>Throughput</b>	I picchi di carico, fino a circa 100(cento) utenti simultaneamente collegati, devono essere gestiti dal sistema senza rallentamenti, garantendo fluidità e una latenza molto bassa. Il sistema, deve garantire consistenza in tutte le operazioni che includono anche chiamate al database, specialmente nei momenti di maggiore carico.

<b>Memoria</b>	Il sistema utilizza un database relazionale per memorizzare tutti i dati. La mole dei dati non rappresenterà un problema di performance del sistema.
----------------	--

### 1.2.2 Criteri di Affidabilità

<b>Robustezza</b>	I componenti devono essere affidabili ed essere in grado di poter mantenere i propri dati anche in caso di guasti come i problemi elettrici, guasti hardware, attacchi informatici, problemi legati al browser.
<b>Disponibilità</b>	Unisa Cinema deve essere disponibile all'uso, 24 ore su 24, da parte degli utenti, grazie ad un server sempre attivo.
<b>Tolleranza all'errore</b>	Il sistema deve essere capace di operare durante condizione d'errore. Ciò sarà reso possibile tra tutte quelle sotto-parti del sistema che hanno un grado di accoppiamento basso, in modo tale che l'errore in un sottosistema non intacchi le funzionalità di un altro. Inoltre, deve essere in grado di pianificare dei backup periodici dell'intero sistema.
<b>Sicurezza</b>	Le tecniche utilizzate, per garantire la sicurezza, sono basate prevalentemente su una 'login', la quale permette il riconoscimento dell'utente. Tutti gli accessi al sistema avvengono tramite un'operazione di routing che controlla se l'utente che sta operando sul sistema ha i permessi adeguati per usufruire di determinate funzionalità. C'è, quindi, un controllo che verifica i permessi degli utenti che verranno adeguatamente distribuiti sulla base del grado di importanza e responsabilità degli Utenti. In particolare un amministratore avrà i massimi permessi poiché ha facoltà di utilizzare qualsiasi tipo di funzione disponibile, mentre un cliente avrà le piene facoltà di gestione del proprio account, quindi delle modifiche del profilo e degli acquisti. Infine un operatore ha le facoltà solo di gestire il proprio account e di visionare gli ordini, visto che il suo ruolo è di minore spicco nel sistema. Inoltre le password sono codificate in MD5 senza essere rese in chiaro in nessun campo di

	nessuna tabella che tiene traccia di queste informazioni all'interno del database.
--	--

### 1.2.3 Criteri di Manutenzione

<b>Estensibilità</b>	La progettazione del sistema sarà condotta in modo da agevolare la facile introduzione di nuove funzionalità utilizzando il linguaggio di markup HTML5, i fogli di stile CSS3, Bootstrap 3.3, JQuery, un template sviluppato con Bootstrap.
<b>Modificabilità</b>	Deve essere possibile intervenire sul codice esistente per correggere eventuali bugs o implementare nuove funzionalità. Bisogna garantire che il codice sia leggibile per rendere agevole la modifica.
<b>Leggibilità</b>	Il codice sarà ben strutturato per semplificare eventuali interventi su di esso.
<b>Tracciabilità dei requisiti</b>	Grazie alla tracciabilità dei requisiti, sarà possibile effettuare le modifiche necessarie al corretto funzionamento del sistema, valutando correttamente i costi e i rischi che le modifiche porteranno.

### 1.2.4 Criteri per l'Utente Finale

<b>Usabilità</b>	Unisa Cinema rende ogni funzione di semplice uso garantendo un'ottima esperienza lavorativa all'utente grazie ad interfacce grafiche intuitive cioè user-friendly.
------------------	--

## 1.3 Definizioni, acronimi e abbreviazioni

- Unisa Cinema: Nome del sistema che verrà sviluppato.
- Operatore: attore del sistema che gestisce l'acquisto del biglietto.
- Ospite: utente non registrato al sistema che può effettuare visualizzazione e ricerche.
- Utente Registrato: utente registrato che può effettuare operazioni di aggiunta, modifica, cancellazione e acquisti.
- DBMS: Database Managment System

## 1.4 Riferimenti

- Bern Bruegge, Allen H. Dutoit, Object-Oriented Software Engineering - Using UML, Patterns, and JAVA, 3rd edition.

## 1.5 Panoramica

Il documento si compone di cinque parti. In particolare, nella prima parte sono stati introdotti gli obiettivi di design e sono stati forniti i riferimenti ad altri materiali.

In seguito, nella sezione sistema software corrente, verrà descritto un sistema software simile.

La sezione sistema software proposto documenta il system design del nuovo sistema. In questa sezione vengono descritti i seguenti elementi:

- Decomposizione in sottosistemi: il sistema viene suddiviso in diversi sottosistemi. Ricordiamo che un sottosistema è formato da un insieme di classi, associazioni, operazioni e vincoli che sono in relazione tra di loro. Ogni sottosistema è caratterizzato dai servizi che offre agli altri sottosistemi.
- Mapping hardware/software: in questa sezione vengono prese decisioni riguardo le piattaforme hardware su cui il sistema dovrà girare, una volta decise le piattaforme è necessario mappare le componenti su di esse.
- Gestione dei dati persistenti: descrive i dati persistenti che vengono memorizzarli dal sistema ed il tipo di infrastruttura usata per memorizzarli.
- Controllo degli Accessi e Sicurezza: descrive il modello degli utenti del sistema in termini di una matrice degli accessi. Questa descrive le informazioni a cui i singoli attori possono accedere e quali sono le operazioni che questi hanno la possibilità di effettuare.
- Controllo Globale del Software: descrive il modo in cui è implementato il controllo globale del software e come si sincronizzano i sottosistemi.
- Condizioni di boundary: in questo paragrafo vengono descritte le condizioni limite del sistema come start-up e shutdown e la gestione dei fallimenti del sistema.

La sezione servizi dei sottosistemi descrive in termini di operazioni quali sono i servizi forniti da ciascun sottosistema.

L'ultima parte del documento è costituita dal glossario che si occupa di elencare una serie di termini e fornire la relativa spiegazione in maniera tale da fornire supporto a coloro che leggono il documento.

## 2. Architettura del Software Corrente

Al giorno d'oggi non esiste un software per la gestione di cinema associativi.

## 3. Architettura del Software Proposto

### 3.1 Panoramica

L'architettura del sistema Unisa Cinema è di tipo client/server. Il server riceve le richieste da parte del 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;

- **Performance:** il client sarà in grado di supportare task interattivi display intensive e il server dovrà fornire operazioni CPU-intensive;
- **Scalabilità:** il server sarà in grado di gestire un grosso numero di client contemporaneamente, grazie alla funzionalità cloud, che utilizza una soluzione Paas.
- **Flessibilità:** per ogni tipologia di utente che effettua l'accesso al sistema, vi sarà un'interfaccia grafica apposita, tramite la quale ogni attore potrà eseguire le operazioni ad esso riservate.
- **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.

Nello sviluppo del nostro 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. In particolare:

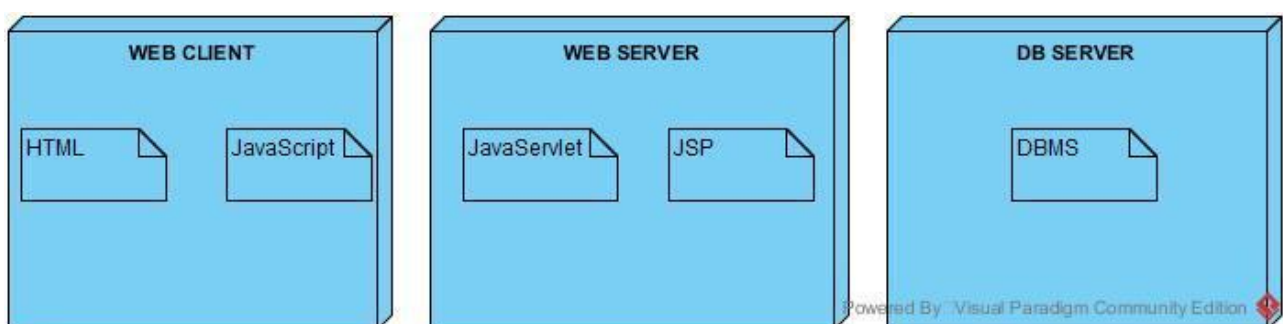
il model si occupa della di gestire i dati e quindi sarà responsabile dell'interazione con il database sottostante

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 dopo aver ricevuto i comandi forniti dall'utente si occuperà di elaborare i dati, passarli al model se necessario e inviare la risposta al view appropriato.

L'utilizzo del modello MVC comporta numerosi vantaggi tra i quali ricordiamo la possibilità di suddividere il lavoro più facilmente tra i vari componenti del team e la maggiore agilità negli interventi di manutenzione. Infatti, l'utilizzo di un modello e di regole standard facilita la comprensione anche chi non ha inizialmente lavorato su quel sottosistema.

All'interno del nostro sistema il model verrà realizzato utilizzando classi Java appropriate, la parte di view verrà implementata utilizzando pagine HTML e JavaScript e i control saranno realizzati tramite Servlet.





### 3.2 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: composto dalle interfacce grafiche e in particolare dai boundary object come le form che vengono compilate dall'utente.
- 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 lo storage layer.
- Storage layer: si occupa della memorizzazione dei dati persistenti e del loro recupero dal database ad esempio eseguendo delle query.

Il sottosistema Presentation Layer è stato suddiviso in quattro sottosistemi:

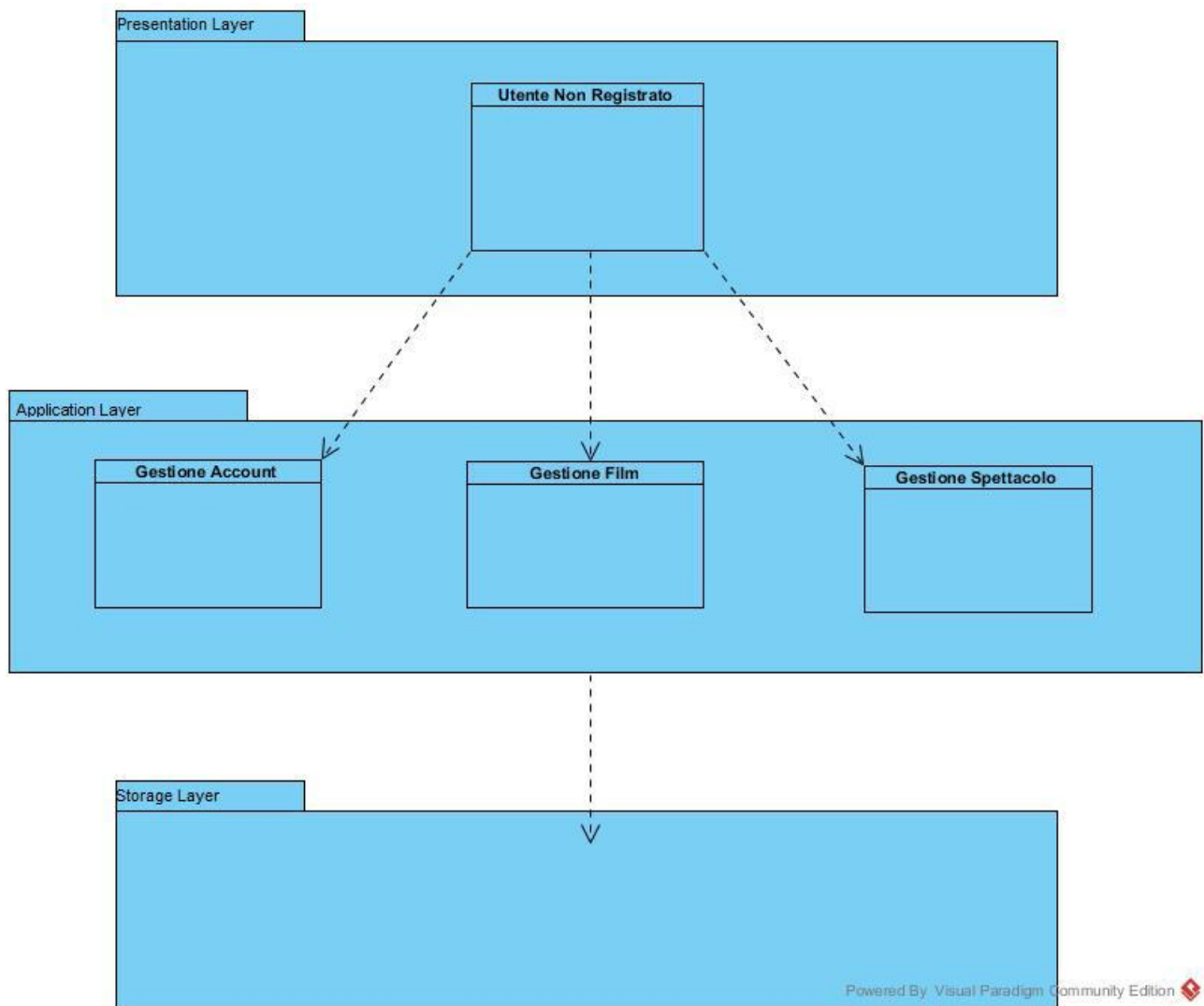
- Sottosistema Cliente. Include tutte le interfacce grafiche a cui il cliente può accedere per visualizzare informazioni relative a film, account, promozioni e spettacoli. Inoltre, può effettuare un acquisto.
- Sottosistema Amministratore. Include tutte le interfacce grafiche a cui l'amministratore può accedere. In particolare, può visualizzare informazioni su Account, Film, Spettacoli, e Promozioni e su quest'ultimi può effettuare anche operazioni di modifica, inserimento e cancellazione.
- Sottosistema Operatore. Include tutte le interfacce a cui l'operatore può accedere per permettergli la visualizzazione delle informazioni del proprio account e la gestione degli Acquisti.
- Sottosistema Utente non Registrato. Include le interfacce accessibili dall'utente non registrato come la pagina di registrazione, visualizzazione delle informazioni su un Film e uno Spettacolo.

Il sottosistema Application Layer è stato diviso in sottosistemi:

- Sottosistema Gestione account, include tutte le operazioni che permettono di visualizzare le informazioni di un account, nel caso di Operatore, Amministratore o Cliente, o crearne uno nuovo, nel caso di un Utente non Registrato.
- Sottosistema Gestione Film, include tutte le operazioni che permettono di visualizzare le informazioni di un film o permettono all'amministratore di inserirne uno nuovo.
- Sottosistema Gestione Spettacolo, include tutte le operazioni che permettono di visualizzare le informazioni di uno spettacolo e che permettono all'amministratore di inserirne uno nuovo o di modificarne uno già esistente.
- Sottosistema Gestione Acquisti, include le funzionalità necessarie al cliente per l'acquisto di un biglietto ed offre all'operatore le informazioni necessarie per gestire l'ordine.
- Sottosistema Gestione Promozione, include tutte le operazioni che permettono di visualizzare le informazioni relative ad una promozione. Inoltre permette all'amministratore di creare una nuova promozione.

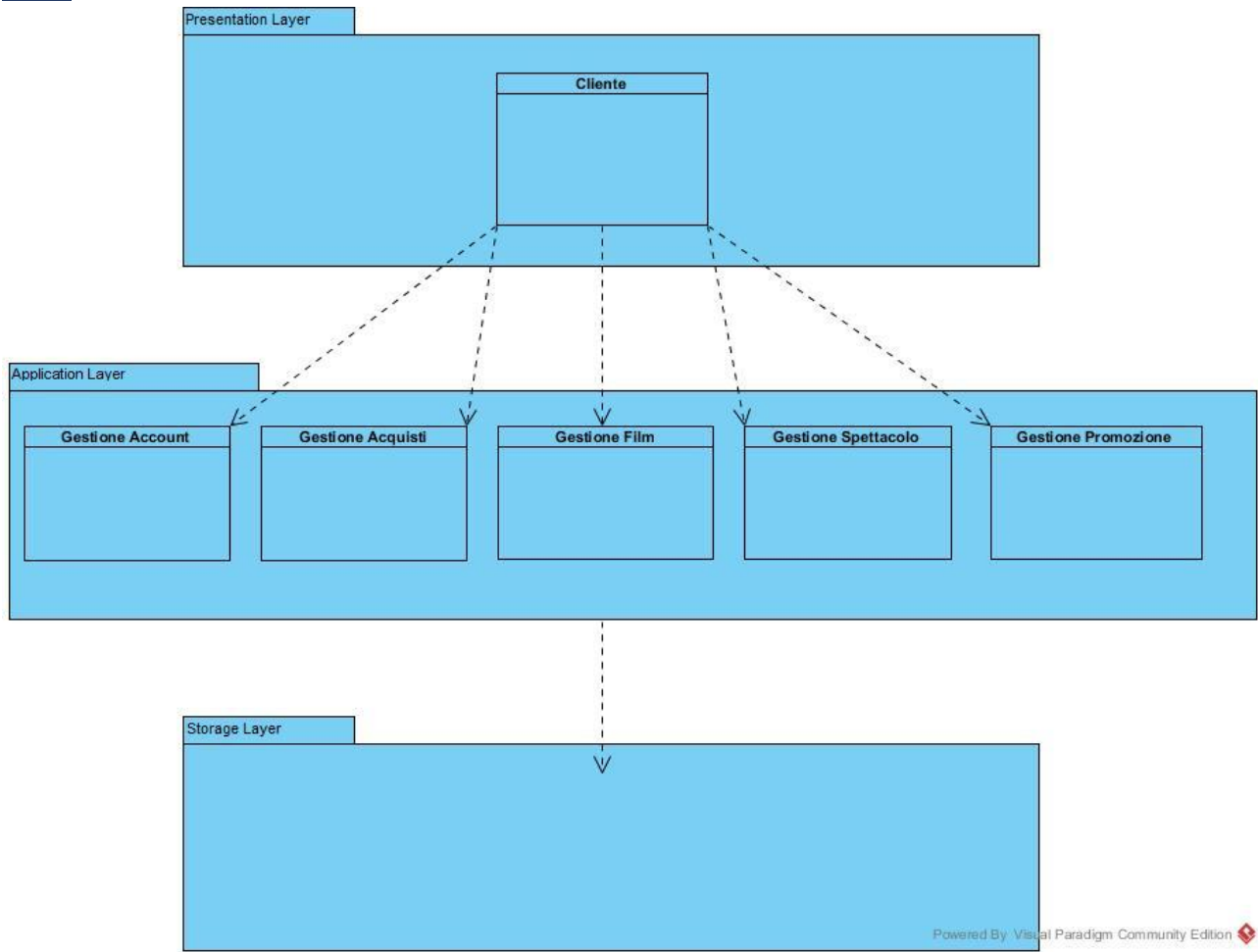
Ora mostriamo nello specifico i vari layer:

### 3.2.1 Utente non registrato



- **Gestione Account:** l'utente non registrato può effettuare la registrazione al sistema inserendo le proprie credenziali.
- **Gestione Film:** l'utente non registrato può ricercare e visualizzare le informazioni relative ai film presenti.
- **Gestione Spettacolo:** l'utente non registrato può visualizzare le informazioni relative agli spettacoli presenti.

### 3.2.2 Cliente



**Gestione account:** il cliente può effettuare l'accesso al sistema utilizzando le proprie credenziali e può effettuare il log out. Inoltre, può visualizzare le proprie informazioni personali.

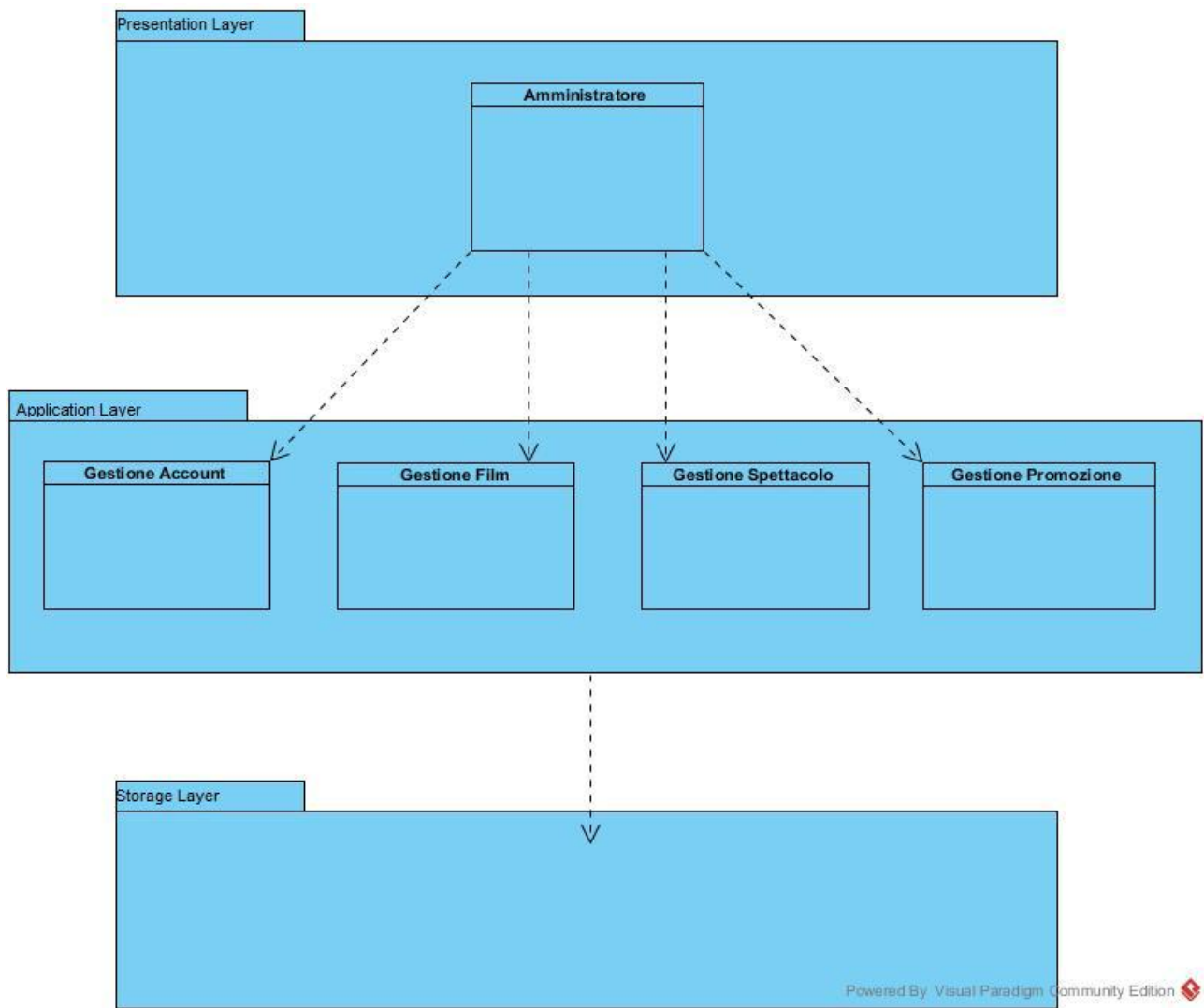
**Gestione acquisti:** il cliente può effettuare l'acquisto di uno o più biglietti relativi ad uno spettacolo.

**Gestione Film:** il cliente può ricercare e visualizzare le informazioni relative ai film presenti.

**Gestione Spettacolo:** il cliente può visualizzare le informazioni relative agli spettacoli presenti.

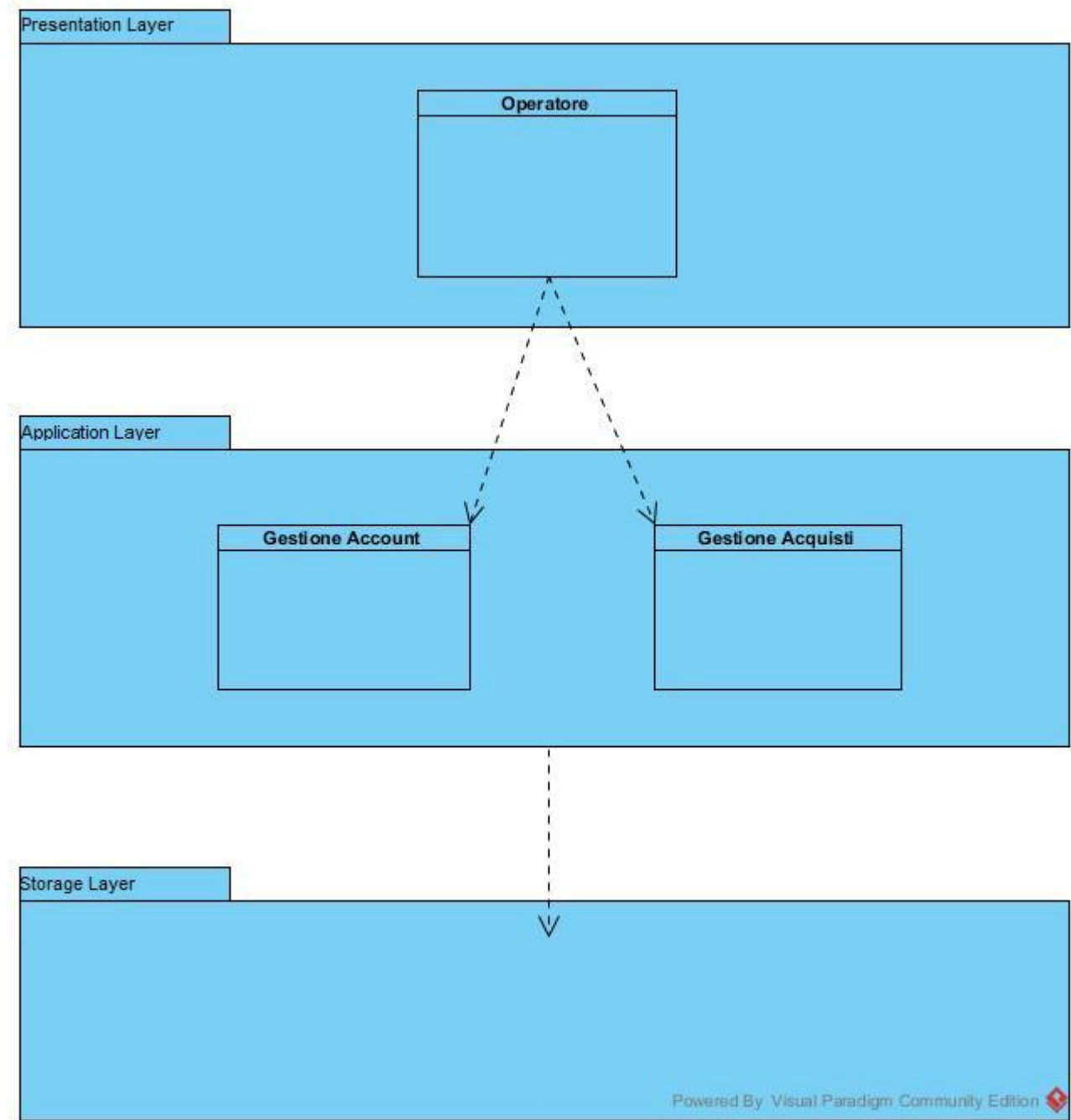
**Gestione Promozione:** il cliente può visualizzare le informazioni relative alle promozioni presenti ed usufruire di esse.

### 3.2.3 Amministratore



- **Gestione Account:** l'utente amministratore può effettuare l'accesso al sistema utilizzando le proprie credenziali e può effettuare il log out. Inoltre, può visualizzare le proprie informazioni personali.
- **Gestione Film:** l'utente amministratore può visualizzare informazioni relative ad un film o crearne uno nuovo.
- **Gestione Spettacolo:** l'utente amministratore può visualizzare e modificare informazioni relative ad uno spettacolo inoltre può creare un nuovo spettacolo relativo ad un film.
- **Gestione Promozione:** l'utente amministratore può visualizzare le informazioni relative ad una promozione ed inoltre può crearne una nuova.

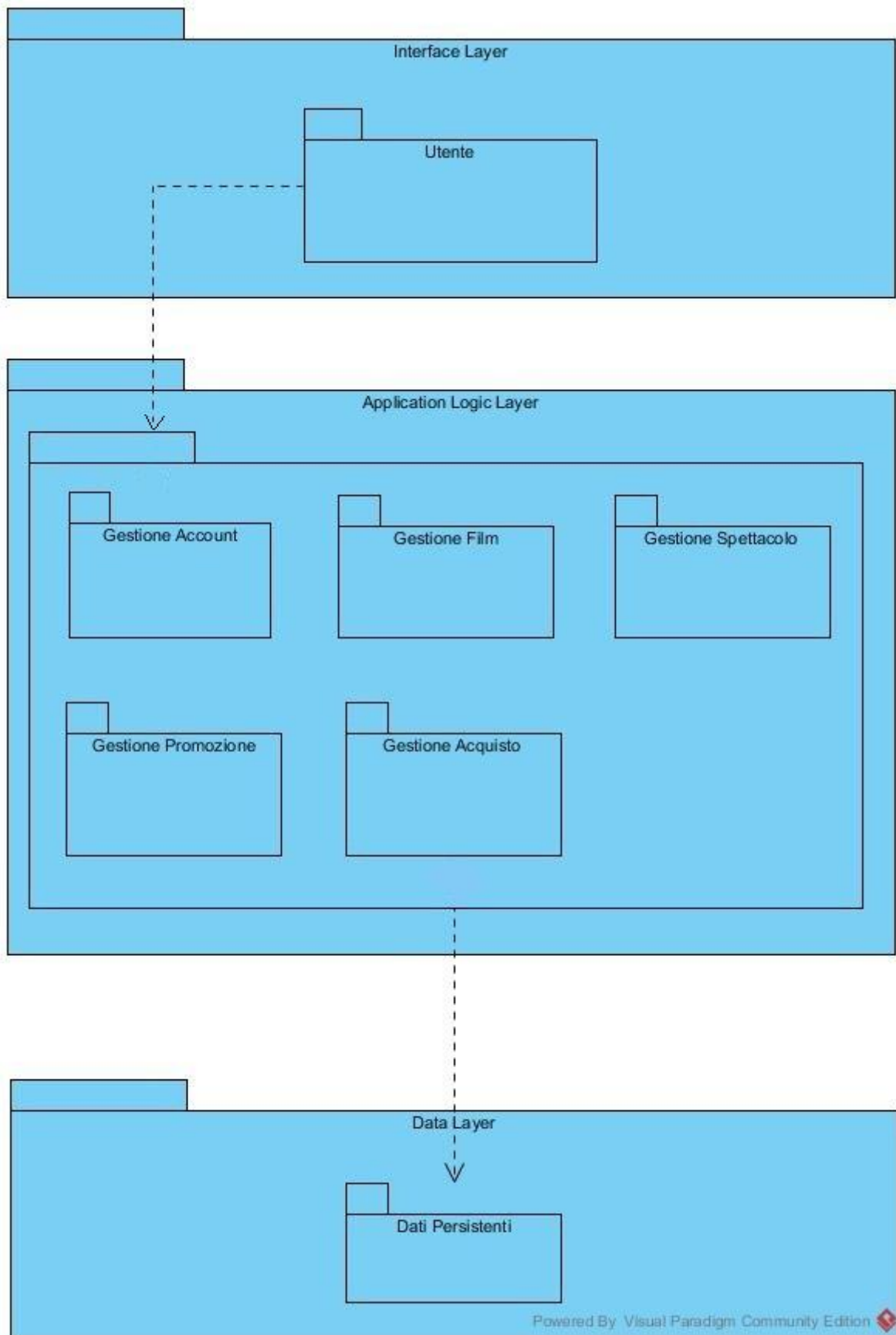
### 3.2.4 Operatore



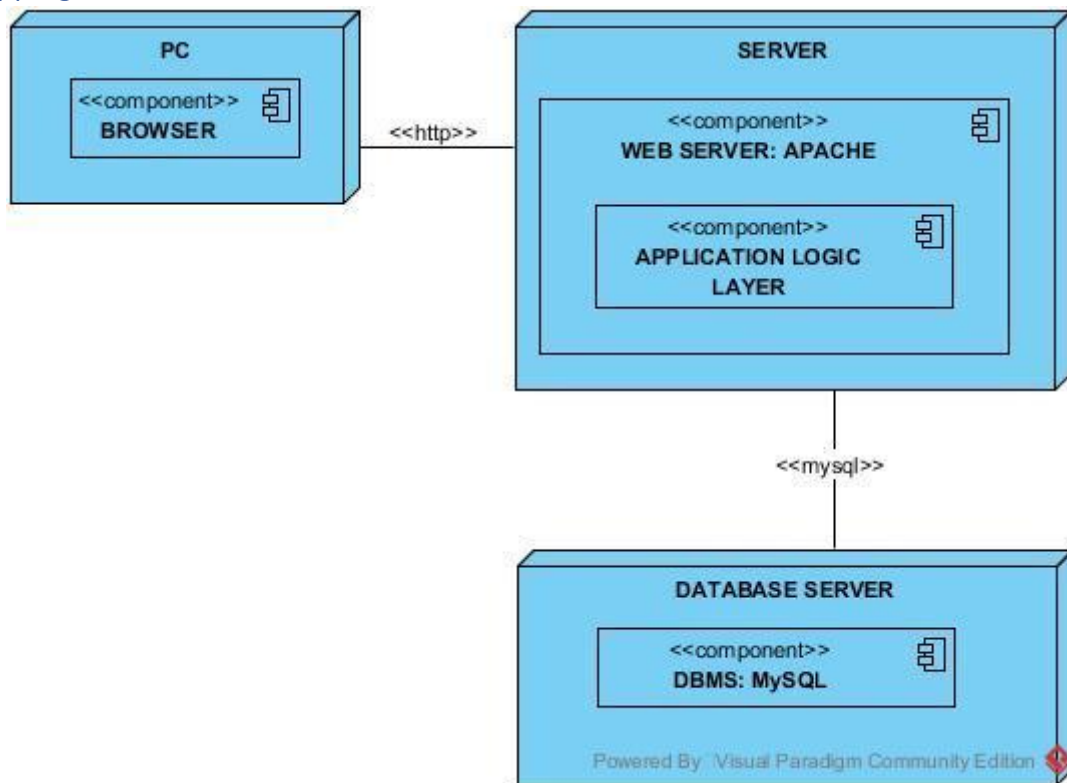
- **Gestione Account:** l'utente operatore può effettuare l'accesso al sistema utilizzando le proprie credenziali e può effettuare il log out. Inoltre, può visualizzare le proprie informazioni personali.
- **Gestione Acquisti:** l'utente operatore può visualizzare i dettagli relativi agli ordini.

### 3.2.5 Diagramma Generale

Viene di seguito riportato un diagramma generale.



### 3.3 Mapping Hardware/Software



#### Web Server

Il server utilizzato è Apache Tomcat versione 8.5.

#### Interface layer

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

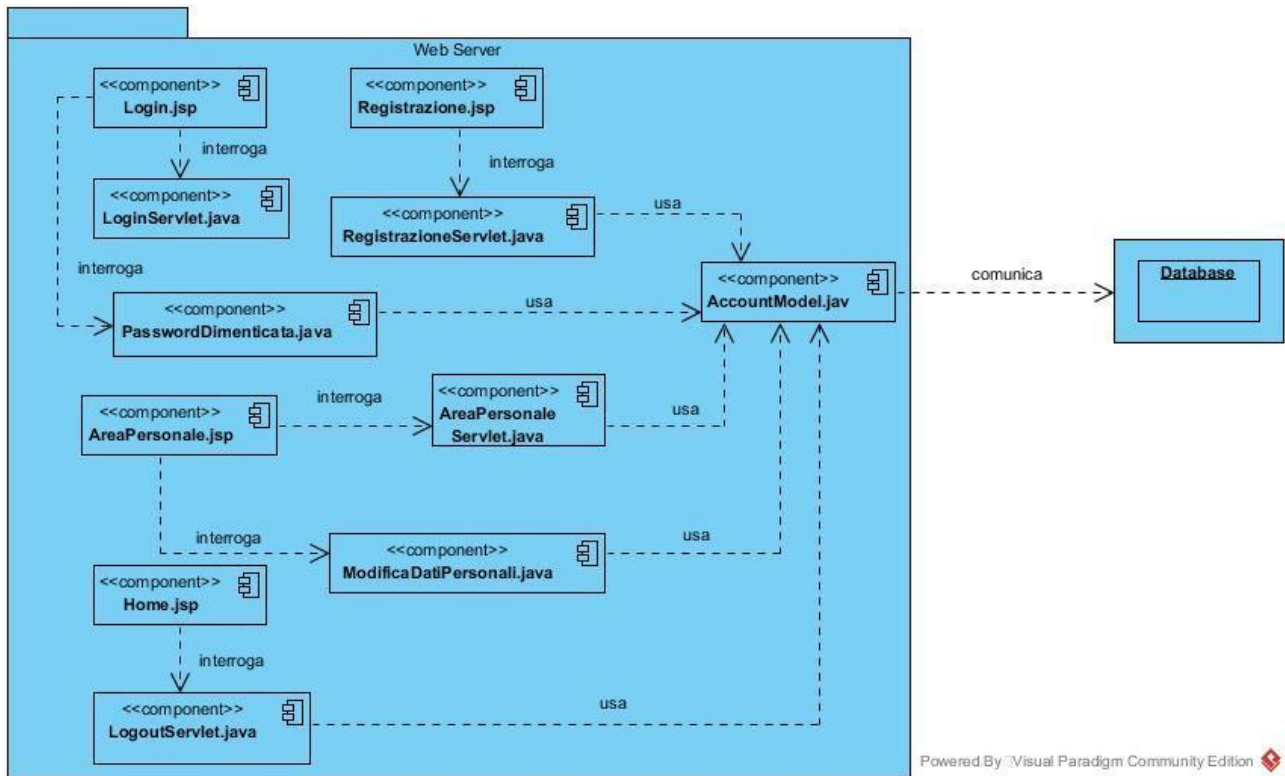
#### Application Logic layer

Il sistema, e quindi le funzionalità, sono implementate in linguaggio JavaServlet. Il codice in JavaServlet verrà tradotto in linguaggio HTML e JSP inoltre il codice risultante viene inviato al browser del client.

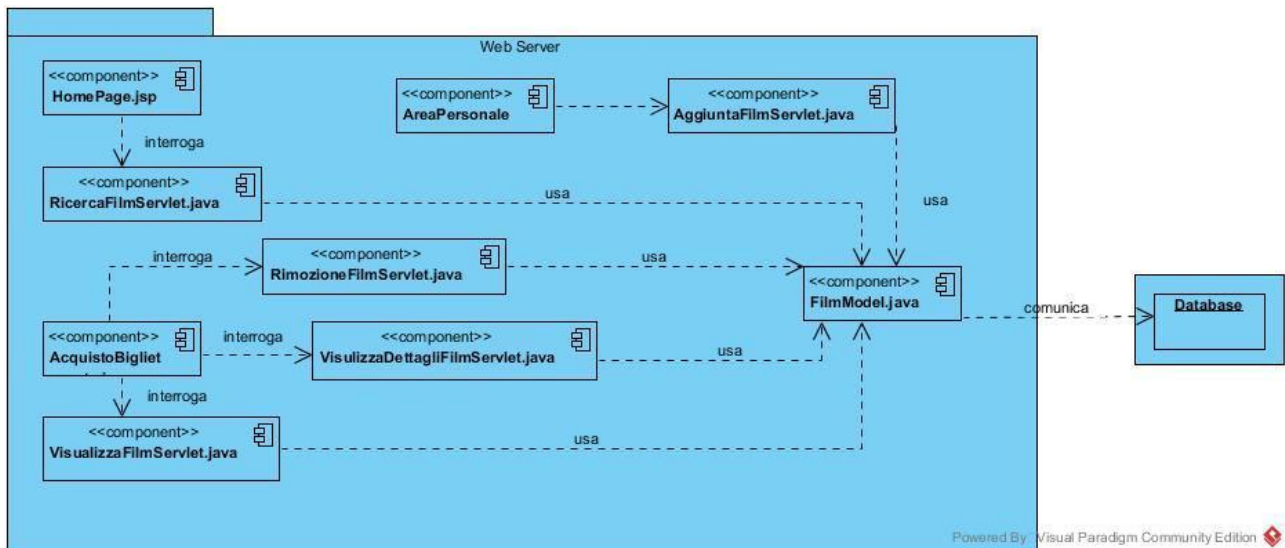
#### Database Server

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

## Gestione Acquisti

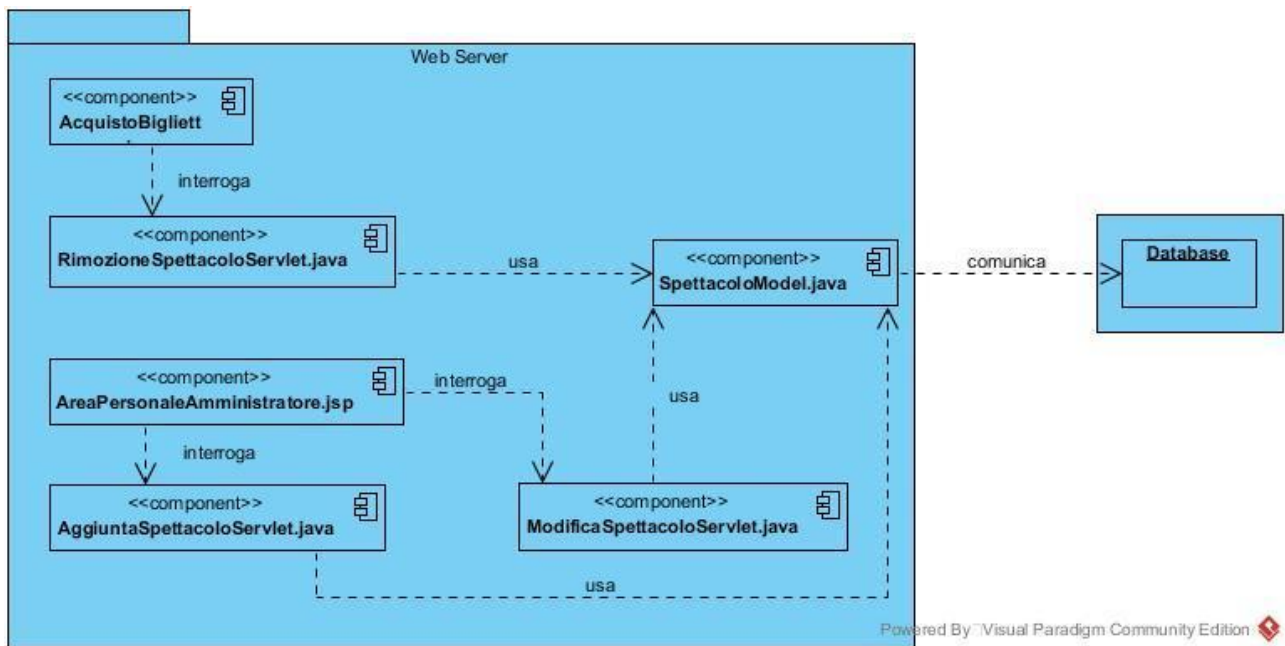


## Gestione Film

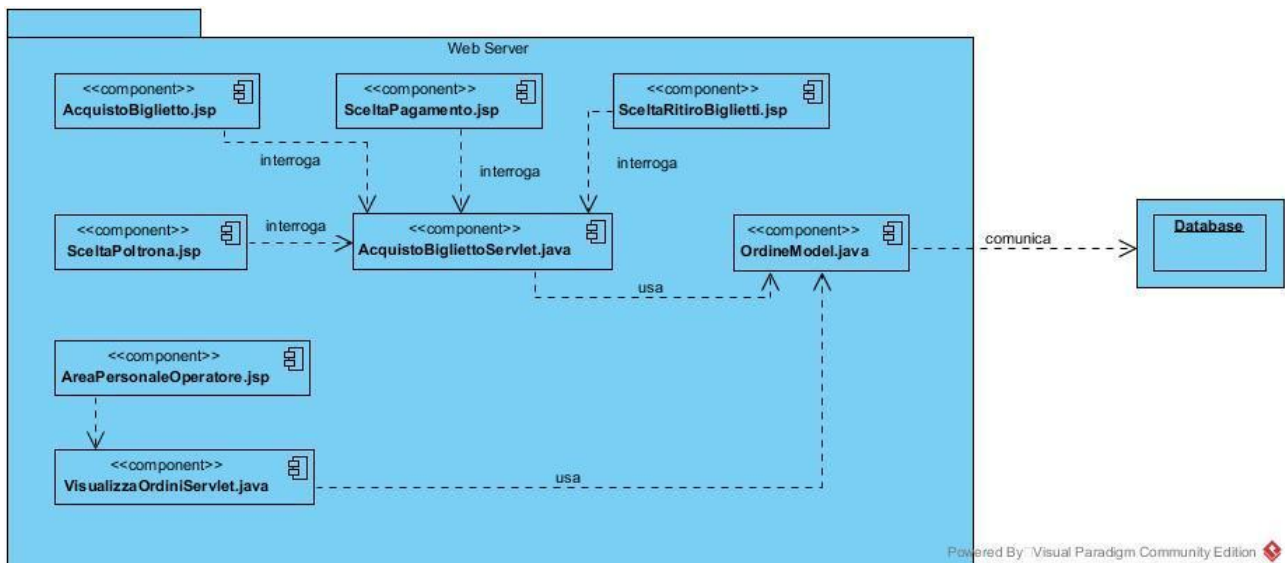




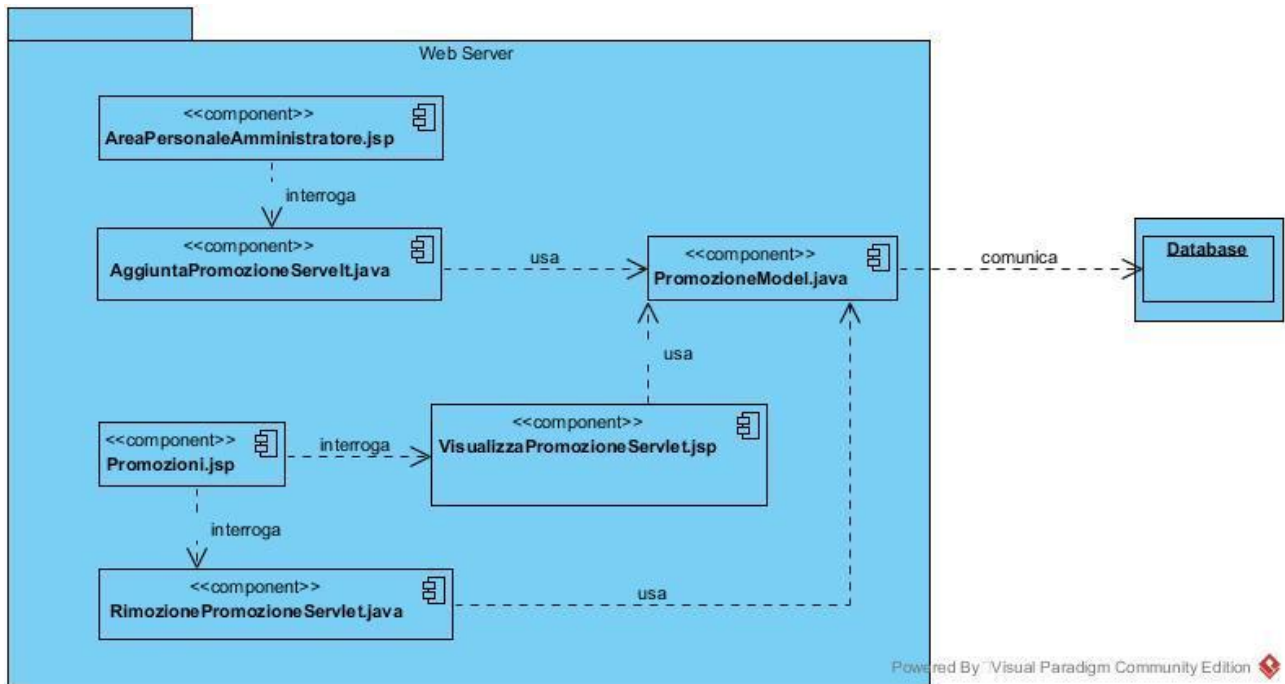
## Gestione Spettacoli



## Gestione Acquisti



## Gestione Promozioni



### 3.4 Gestione dei dati persistenti

Si rimanda al documento "Unisa Cinema – Gestione Dati Persistenti "allegato al presente.

### 3.5 Controllo degli accessi e della sicurezza

Unisa Cinema è un sistema multi-utente, ci sono diversi attori che 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 compatta e dettagliata grazie ad una matrice degli accessi riportata di seguito:

Oggetti	Attori	Amministratore	Cliente	Operatore
Account		✓ Login ✓ Logout ✓ Visualizza Profilo	✓ Login ✓ Logout ✓ Visualizza Profilo ✓ Registrazione ✓ Modifica Profilo ✓ Password Dimenticata	✓ Login ✓ Logout ✓ Visualizza Profilo
Film		✓ Ricerca Film ✓ Visualizza Film ✓ Visualizza Dettagli Film ✓ Aggiunta Film ✓ Rimozione Film	✓ Ricerca Film ✓ Visualizza Film ✓ Visualizza Dettagli Film	✓ Ricerca Film ✓ Visualizza Film ✓ Visualizza Dettagli Film
Spettacoli		✓ Aggiunta Spettacolo ✓ Rimozione Spettacolo ✓ Modifica Informazioni Spettacolo		
Acquisto			✓ Acquisto Biglietto	✓ Visualizza Ordini Effettuati
Promozione		✓ Visualizza Promozione ✓ Aggiunta Promozione ✓ Rimozione Promozione		

### 3.6 Controllo del software globale

Il controllo del flusso software viene gestito da classi `JavaServlet` che interagendo con il client, il quale si interfaccia tramite un web browser, svolgono le varie operazioni. Il server smista ogni nuova richiesta alle classi `JavaServlet`, inoltrando poi la risposta al client.

### 3.7 Condizione boundary

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.

#### 3.7.1 Avvio del sistema

Per lo start-up del sistema è necessario avviare il web server. In seguito, tramite l'interfaccia di Login, sarà possibile autenticarsi tramite opportune credenziali (username e password) come Amministratore, Operatore e Cliente.

#### 3.7.2 Terminazione del sistema

Al momento della corretta chiusura dell'applicazione, si ha la terminazione del sistema con un regolare Logout. Per terminare il server è necessario l'intervento dell'amministratore. Questo dovrà effettuare la procedura di terminazione dopo la quale non sarà possibile per nessun client connettersi al sistema

#### 3.7.3 Fallimento del sistema

Nel caso si verifichi un errore dovuto all'hardware o al software si cercherà di ripristinare una configurazione del sistema precedente allo stato d'errore. Visto che i dati sono gestiti dal DBMS non c'è rischio di perderli. Tuttavia, se si verifica un guasto al supporto di memorizzazione si può avere una perdita di dati.

### Scenari

<b>Nome Scenario</b>	Startup Sistema
<b>Istanze di Attori Partecipanti</b>	Paolo: Amministratore
<b>Flusso di Eventi</b>	<ol style="list-style-type: none"><li>1. Paolo decide di voler avviare il sistema e quindi clicca sul pulsante "Avvia".</li><li>2. Il sistema, con le opportune procedure di avvio, attiva i server e i relativi servizi in remoto rendendosi disponibile ad eventuali richieste.</li><li>3. Il sistema notifica il successo della procedura.</li></ol>

<b>Nome Scenario</b>	Shutdown Sistema
<b>Istanze di Attori Partecipanti</b>	Paolo: Amministratore

<b>Flusso di Eventi</b>	<ol style="list-style-type: none"> <li>1. Paolo decide di voler arrestare il sistema e quindi accede alla pagina dedicata e clicca sul pulsante "Arresta".</li> <li>2. Il sistema effettua una scansione per verificare se ci sono ancora richieste in sospeso.</li> <li>3. Il sistema porta a termine le eventuali richieste in sospeso</li> <li>4. Tramite le opportune procedure di arresto il sistema disattiva i servizi in remoto e il server.</li> <li>5. Il sistema notifica il successo della procedura.</li> </ol>
-------------------------	--

### Casi d'uso

<b>ID</b>	UC_Startup
<b>Nome Caso Uso</b>	Startup Server
<b>Istanze di Attori Partecipanti</b>	Amministratore
<b>Condizione di Entrata</b>	L'amministratore visualizza la console per effettuare l'avvio del sistema.
<b>Flusso di Eventi</b>	<div> <div> <p>Utente</p> <p>L'amministratore clicca sul pulsante "Avvia"</p> </div> <div> <p>Sistema</p> <p>Il sistema attiva i propri servizi rendendosi disponibile in</p> </div> </div>

	tutte le sue funzionalità.
<b>Condizione di Uscita</b>	Il server è attivo e i relativi servizi sono disponibili
<b>Eccezioni</b>	Errore Startup.

<b>ID</b>	UC_Shutdown
<b>Nome Caso Uso</b>	Shutdown Server
<b>Istanze di Attori Partecipanti</b>	Amministratore
<b>Condizione di Entrata</b>	L'amministratore visualizza la console per effettuare la terminazione del sistema
<b>Flusso di Eventi</b>	<div> <div> <p>Utente</p> <p>L'amministratore clicca sul pulsante "Spegni".</p> </div> <div> <p>Sistema</p> <p>Il sistema effettua la procedura per la terminazione del server.</p> </div> </div>
<b>Condizione di Uscita</b>	Il server si è spento correttamente
<b>Eccezioni</b>	Errore shutdown.

## 4. Servizi dei sottosistemi

### 4.1 Gestione Account

<b>Sottosistema</b>	Gestione Account
<b>Descrizione</b>	Sottosistema che gestisce e raccoglie tutte le funzionalità di cui possono usufruire gli utenti del sistema ovvero: registrazione di un cliente, autenticazione degli utenti, la visualizzazione dei propri profili da parte degli utenti.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
Login	Questa funzionalità permette di effettuare l'accesso al sistema
Logout	Questa funzionalità permette di effettuare la disconnessione dal sistema.
Visualizza Profilo	Questa funzionalità permette al cliente di visualizzare le informazioni inerenti al proprio profilo.
Registrazione	Questa funzionalità permette ad un utente non registrato di registrarsi al sistema
Modifica Profilo	Questa funzionalità permette al cliente di modificare i dati personali.
Password Dimenticata	Questa funzionalità permette al cliente di modificare la password qualora quest'ultima venga dimenticata.

### 4.2 Gestione Film

<b>Sottosistema</b>	Gestione Film
<b>Descrizione</b>	Sottosistema che gestisce e comprende tutte le funzionalità di un film, dove un utente visualizza i dettagli di un film, l'admin può aggiungerli o cancellare i film e quindi operare su di essi.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
Ricerca Film	Questa funzionalità permette a tutti gli utenti di ricercare un film.
Visualizza Film	Questa funzionalità permette di visualizzare l'elenco dei film.
Visualizza Dettagli Film	Questa funzionalità permette di visualizzare tutti i dettagli inerenti ai film.
Aggiunta Film	Questa funzionalità permette ad un amministratore di aggiungere un film.
Rimozione Film	Questa funzionalità permette ad un amministratore di rimuovere un film.



### 4.3 Gestione Spettacolo

<b>Sottosistema</b>	Gestione Spettacolo
<b>Descrizione</b>	Sottosistema che comprende tutta la gestione degli spettacoli, associati ad un film e permette ad un amministratore di aggiungere, cancellare e modificare le informazioni di uno spettacolo.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
Aggiunta Spettacolo	Questa funzionalità permette ad un amministratore di aggiungere uno spettacolo inerente ad un film.
Rimozione Spettacolo	Questa funzionalità permette all'amministratore di rimuovere uno spettacolo inerente ad un film
Modifica Informazioni Spettacolo	Questa funzionalità permette all'amministratore di modificare le informazioni inerenti ad uno spettacolo (film, orario di proiezione e sala di proiezione).

### 4.4 Gestione Acquisto

<b>Sottosistema</b>	Gestione Acquisto
<b>Descrizione</b>	Sottosistema che comprende le operazioni di acquisto di un biglietto da parte di un cliente ed inoltre scegliere in essi il pagamento, il ritiro dei biglietti e anche la poltrona. L'operatore può visualizzare gli ordini effettuati.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>
Acquisto Biglietto	Questa funzionalità permette ad un cliente di prenotare un o più biglietti, scegliere il metodo di pagamento, inoltre se stampare i biglietti da casa oppure ritirarli al botteghino, e in seguito di scegliere la/le poltrona/e che si preferisce, e alla fine di visualizzare il riepilogo dell'ordine.
Visualizza ordini effettuati	Questa funzionalità permette all'operatore di visionare gli ordini effettuati dai Clienti.

### 4.5 Gestione Promozione

<b>Sottosistema</b>	Gestione Promozione
<b>Descrizione</b>	Sottosistema che permette all'amministratore di gestire le promozioni che sono disposte periodicamente.
<b>Servizi Offerti</b>	
<b>Servizio</b>	<b>Descrizione</b>

Visualizza Promozione	Questa funzionalità permette al cliente di visualizzare le promozioni disponibili.
Aggiunta Promozione	Questa funzionalità permette all'amministratore di aggiungere una nuova promozione.
Modifica Promozione	Questa funzionalità permette all'amministratore di rimuovere una o più promozioni.

## 5.GLOSSARIO

**Client:** componente che accede ai servizi o alle risorse di un'altra componente, detta server.

**Deployment Diagram:** Schema che descrive la struttura dinamica del sistema

**DBMS:** programma informatico (o, più frequentemente, un insieme di programmi) progettato per gestire un database, ovvero un insieme di numerosi dati strutturati. Le operazioni, normalmente, sono richieste da un gran numero di utenti.

**Form:** finestra di dialogo incorporata in una pagina Web che consente all'utente di inserire informazioni destinate ad un server. Generalmente richiede un programma sul server che si occupi di esaminare le informazioni inviate. E' composto da spazi (campi) predefiniti, ad esempio menù a tendina, elenchi puntati o caselle di testo libero.

**JDBC:** API per il linguaggio di programmazione Java che serve ai client per connettersi a un database. Fornisce metodi per interrogare e modificare i dati. È orientata ai database relazionali.

**Login:** Procedura attraverso la quale ci si collega con un qualsiasi servizio in linea. All'utente viene assegnato un nome di login ed una password che vengono richiesti dal sistema ogni volta che ci si collega.

**Layer:** E' un insieme di classi con funzionalità simile (tipicamente raggruppati in un unico package).

**Logout:** Operazione attraverso la quale si termina un collegamento con un sistema al quale si ha accesso attraverso un nome utente e una password (vedi login).

**MySQL:** Database management system relazionale, composto da un client con interfaccia a caratteri e un server, disponibile su molte piattaforme.

**Password:** È un metodo di sicurezza che, mediante una stringa di caratteri, permette di identificare un utente specifico. Generalmente le password sono formate da una sequenza di lettere e numeri; digitando correttamente questi caratteri, si può avere accesso al computer o alla rete.

**Package:** Un Package rappresenta una collezione di classi ed interfacce che possono essere raggruppate in base alla funzione comune da esse svolta

**Server:** Programma di gestione di un servizio che invia informazioni in un particolare formato ricevuto e interpretato da un programma Client dal lato ricevente.

**Shutdown:** fase di terminazione del sistema.

**Start-up:** processo di accensione e di avvio di un computer, di un dispositivo o di un sistema.