|  |
| --- |
| **Università degli Studi di Salerno Corso di Ingegneria del Software** |

**MusicParadise  
System Design Documentation  
Versione 0.1**

**LOGO PROGETTO**

****

Data: 20/11/2017

**Coordinatore del progetto:**

|  |  |
| --- | --- |
| Nome | Matricola |
|  |  |
|  |  |

**Partecipanti:**

|  |  |
| --- | --- |
| Nome | Matricola |
| **Antonio Spera** | 0512103538 |
| **Alessandro De Riso** | 0512103802 |
| **Domenico Pannone** | 0512103730 |
| **Vincenzo Pandolfo** | 0512103694 |
|  |  |
|  |  |

|  |  |
| --- | --- |
| **Scritto da:** |  |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versione | Descrizione | Autore |
| 01/12/2017 |  | Inserimento condizioni limite del sistema(Accensione, spegnimento, crash) | Alessandro De Riso |
| 07/12/2017 |  | Inserimento ER | Domenico Pannone |
| 08/12/2017 |  | Revisione ER | Antonio Spera |
| 11/12/2017 |  | Controllo accessi e sicurezza | Vincenzo Pandolfo |
| 15/12/2017 |  | Correzione e revizione controllo accessi e sicurezza, aggiunta tabelle DB | Antonio Spera |
|  |  |  |  |

Sommario

[1. Introduzione 4](#_Toc501618708)

[1.1 scopo del sistema 4](#_Toc501618709)

[1.2 Obiettivi di design 5](#_Toc501618710)

[1.3 definizioni, acronimi, e abbreviazioni 6](#_Toc501618711)

[1.4 riferimenti 6](#_Toc501618712)

[2 Architettura software corrente 7](#_Toc501618713)

[3 Architettura software proposta 8](#_Toc501618714)

[3.1 Overview 8](#_Toc501618715)

[3.2 Decomposizione sottosistemi 9](#_Toc501618716)

[3.3 Mapping HardWare / Software 10](#_Toc501618717)

[3.4 Gestione dei dati persistenti 11](#_Toc501618718)

[3.5 Controllo accesso e sicurezza 11](#_Toc501618719)

[3.6 Controllo globale sotware 12](#_Toc501618720)

[3.7 Boundary conditions 12](#_Toc501618721)

[4 Servizi sottosistemi 14](#_Toc501618722)

[5 Glossario 16](#_Toc501618723)

# Introduzione

## scopo del sistema

Si vuole realizzare un software, chiamato MusicParadise.com , per la gestione di un sito e-commerce destinato alla vendita prodotti musicali destinati a musicisti professionisti oppure a semplici appassionati. A tal proposito, possiamo notare che un articolo musicale, ossia un prodotto, può essere acquistato da più clienti e, viceversa, che un cliente può acquistare più prodotti. I prodotti sono caratterizzati da un nome, dalla marca, dal tipo di prodotto, dal prezzo, da foto e da una breve descrizione. Inoltre, un prodotto, potrebbe essere messo in offerta ed avere, quindi, un prezzo scontato. Un visitatore, che si collega al sito di e-commerce mediante questo software, può registrarsi al sito e diventare un potenziale cliente. Un cliente, è caratterizzato da un nickname che lo identifica univocamente, dal nome, dal cognome, da un numero di telefono, da un’email, da una password e se il nostro cliente sarà una ditta ci sono dei campi facoltativi da riempire “ditta” e “Partita IVA”. Oltre al cliente abbiamo anche dei gestori che si occupano della gestione delle varie aree del sito. Ci saranno diversi tipi di gestori :

* Gestore offerte: Visualizzare offerte attive e inserirne altre, modificare alcune già esistenti e eliminarle.
* Gestore prodotti: Visualizzare i prodotti nel sistema, aggiungerne altri, modificarli o eliminarli.
* Gestore ordini: Visualizza tutti gli ordini effettuati dai clienti e modifica il suo stato.
* Gestore utenti: Visualizza tutti i clienti registrati al sito e ha la possibilità di bannare alcuni.

Il software dovrà consentire ai clienti di: autenticarsi, effettuare acquisti, tenere traccia degli acquisti effettuati, tenere traccia dei prodotti scelti per un eventuale acquisto e eliminare uno o più prodotti dalla lista dei prodotti scelti. Il software, inoltre, deve consentire ai clienti/visitatori la ricerca prodotti tramite il loro nome, la loro marca oppure il loro modello. Ogni prodotto visualizzato dal sito di e-commerce, tramite il supporto del software, avrà anche delle foto ad esso associato. In particolare, un prodotto ha una o più foto che permette di visualizzare il prodotto in modo dettagliato.

## Obiettivi di design

**Criteri di performance**

**Tempo di risposta**

* Per il Login il tempo di risposta dovrà essere circa 2 secondi;
* Per la visualizzazione di un prodotto il tempo di risposta dovrà essere circa 1 secondo;
* Per il completamento di un acquisto il tempo di risposta dovrà essere massimo 30 secondi;

**Throughput**

Il sistema non ha grossi vincoli per portare a compimento i task; l’unica limitazione potrebbe essere dovuta alla performance del database utilizzato, al tipo di web server e dalla latenza della rete.

**Memoria**

Il sistema richiede lo spazio necessario per supportare il web server e lo spazio alla memorizzazione e all’archiviazione dei dati nell’unico database presente. In cui sono memorizzati tutti i prodotti, gli utenti, gli ordini, le offerte.

**Criteri di affidabilità**

**Affidabilità**

Il sistema deve essere in grado di effettuare le operazioni nella loro completezza, lasciando così i valori dei dati o nella situazione precedente ad un crash, o comunque in uno stato consistente. In caso di malfunzionamenti della rete, gli utenti devono attendere la risoluzione di questi, e quindi il ripristino della rete per inoltrare le richieste al server. Inoltre il sistema deve essere disponibile 24h al giorno 7/7 giorni, salvo aggiornamenti o malfunzionamenti del server. Gli aggiornamenti del server, preferibilmente, dovranno essere effettuati nelle ore notturne.

**Sicurezza**

Ogni cliente potrà effettuare l’autenticazione al sito inserendo una username e una password. Lo stesso potrà fare ogni gestore ma verranno reindirizzati alla loro pagina dedicata.

I dati sensibili devono essere memorizzati all'interno di un database, accessibile tramite delle credenziali dagli admin.

**Robustezza**

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

**Criteri di manutenzione**

**Estendibilità**

Grazie al linguaggio di programmazione usato(“JAVA”) che risulta essere molto portabile ed estendibile,

diventerà molto facile estendere classi esistenti e aggiungere nuove funzionalità al sistema. E’ possibile,

quindi, sfruttare il riuso per estendere le potenzialità del sistema.

**Adattabilità**

Il sistema può essere facilmente riadattato ad un nuovo dominio con semplici modifiche al sistema.

**Portabilità**

Le tecnologie impiegate garantiscono una totale indipendenza, rendendo così possibile utilizzare la stessa applicazione web su qualsiasi piattaforma che supporta Java (Windows, Unix, Solaris, Macintosh).

## definizioni, acronimi, e abbreviazioni

**RAD**: Requirements Analysis Document.

**SDD**: System Design Document.

**DB**: Database.

## riferimenti

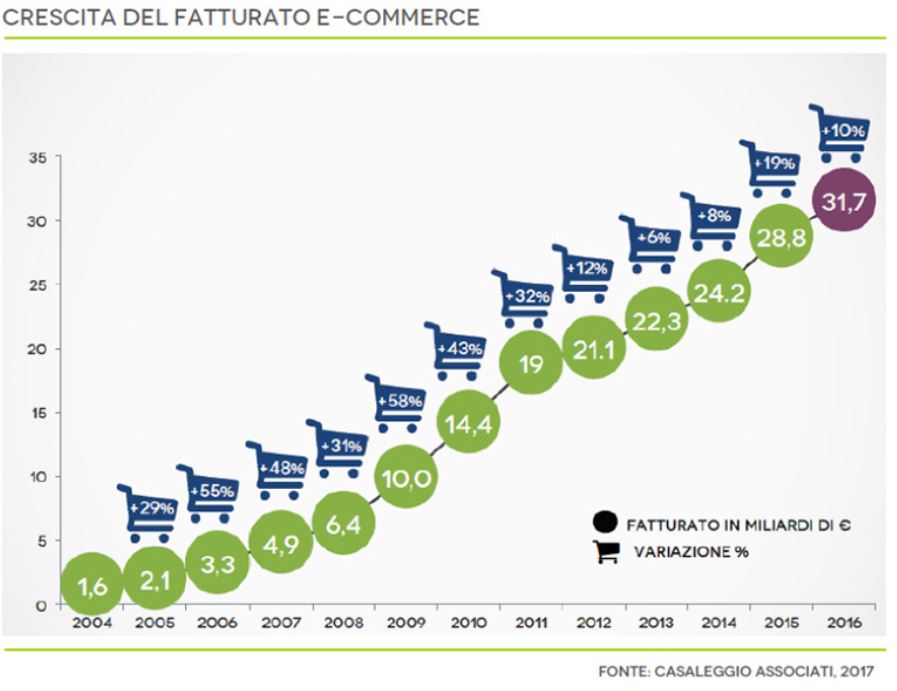
In questo documento abbiamo riferimenti a:

* Rad;

# Architettura software corrente

Il sistema che si vuole implementare non sostituisce nessun sistema preesistente, per questo motivo abbiamo deciso di formulare un’analisi dei sistemi concorrenti e analizzare i vantaggi che un e-commerce ossa portare alla nostra attività.

Analizzando alcune statistiche, abbiamo scoperto che lo sviluppo di un sistema di e-commerce porterebbe un aumento del fatturato nelle piccole attività, come si evince dal grafico seguente, dal 2004 il fatturato ottenuto dal commercio elettronico è aumento circa 30 volte, questo ci fa capire che investire in questo campo, abbiamo buone possibilità di comunicare e vendere i nostri prodotti in tutto il mondo.



# Architettura software proposta

## Overview

Per la decomposizione in sottosistemi abbiamo utilizzato la tecnica della stratificazione (layering) utilizzando il 3-layer. La stratificazione ci ha permesso l’individuazione dei seguenti 3 livelli:



**Presentation Layer (PL)**: è utilizzato come layer per ricevere o mostrare dati al cliente, quindi è il primo step (in caso di input del client) o l’ultimo step (in caso di output visivo) del workflow di una richiesta client. È in oltre definito come “interfaccia con l’utente“.

**Business Logic Layer (BLL)**: è la logica che utilizza i dati ricevuti o acquisiti per produrre un risultato.

**Data Access Layer (DAL)**: Solitamente è il database, ma genericamente è un sistema di salvataggio persistente dei dati, con i vincoli di integrità e di sicurezza.

## Decomposizione sottosistemi

I sottosistemi individuati sono:

**Data layer composto da 1 sottosistema**

1. Storage

**Business logic composto da 5 sottosistemi**

1. Accesso
2. Gestione profilo
3. Acquisto
4. Gestione catalogo
5. Gestione offerte
6. Gestione utenti
7. Gestione ordini
8. Amministrazione

**Presentation layer composto da 1 sottosistema**

1. Interfacce

Immagine che contiene screenshot

Descrizione generata con affidabilità molto elevata

## Mapping HardWare / Software

Il sistema musicParadise.com è stato mappato cosi:

Immagine che contiene screenshot

Descrizione generata con affidabilità molto elevata

Il sistema è composto da due nodi il **client** e **server.** Il **client** l’utente accede da remoto al sistema attraverso il web broswer. Il **server** contiene il **web server** su cui viene installato il sistema e dal quale vengono erogati i servizi. Il protocollo di comunicazione è l’**HTTTP.**

L’utente accede al sistema dalla propria macchina utilizzando un broswer. Sul lato **server** risiederanno le **JSP, servlet** e il database fisico.

Il livello presentation comprende la visualizzazione delle informazioni all’utente, attraverso qualsiasi broswer. Per i boundary verrà utilizzata la tecnologia delle jsp. Abbiamo due tipi di visualizzazione quella **statica** e quella **dinamica:**

* **Statica:** sono pagine di informazioni ad esempio la pagina dei contatti, in generale si intende tutte quelle pagine che non richiedono l’elaborazione del server.
* **Dinamiche:** ad esempio pagine di ricerca, di inserimento di dati, di autenticazione, di registrazione. Quindi in questo caso si intende qualsiasi pagina che richiede l’elaborazione.

I control vengono realizzati tramite **servlet,** si occupa dell’elaborazioni delle informazioni e operazioni di interazione con il data accesso layer. Il data access layer gestisce la persistenza dei dati interfacciandosi con la base di dati, verrà utilizzato **MySql.**

## Gestione dei dati persistenti

Il sistema MusicParadise utilizzerà un database di tipo relazionale, per permettere la gestione e memorizzazione permanente di un grande insieme di dati che possono essere acceduti da utenti e applicazioni diverse. Altri dettagli sono riportate nella **documentazione del database.**

## Controllo accesso e sicurezza

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sottosistemi  Attore | Accesso | Profilo | Acquisto | Ricerca | Gestione-Prodotto | Gestione-offerte | Gestione-utenti | Gestione-ordine | Amministrazione |
| Cliente | * Login * Registrazione * Logout | * Modifica * Visualizza | * Aggiungi prodotto al carrello * Elimina prodotto dal carrello * Acquista | * Ricerca prodotto |  |  |  |  |  |
| Amministratore | * Login * Logout |  |  |  |  |  |  |  | * Aggiungi gestore * Elimina gestore |
| Gestore-ordini | * Login * Logout |  |  |  |  |  |  | * Visualizza ordine * Modifica stato ordine |  |
| Gestore-Prodotti | * Login * Logout |  |  |  | * Aggiungi prodotto * Modifica prodotto * Elimina prodotto |  |  |  |  |
| Gestore-Offerte | * Login * Logout |  |  |  |  | * Aggiungi offerta * Modifica offerta * Elimina offerta |  |  |  |
| Gestore-Utenti | * Login * Logout |  |  |  |  |  | * Visualizza * Elimina utente |  |  |

## Controllo globale sotware

Il sistema prevede la gestione del flusso di controllo di tipo esplicito, i controlli vengono gestiti event-driven

## Boundary conditions

|  |  |
| --- | --- |
| **Nome:** | **Avvio del server** |
| **Partecipante:** | Amministratore del sito |
| **Flusso degli eventi:** | 1. L’amministratore avvia la macchina su cui è installato il server; 2. L’amministratore manda in esecuzione il monitore del server; 3. Il sistema mostra l’icona del monitor; 4. L’amministratore avvia il server; 5. Il Sistema controlla che il server sia stato chiuso precedentemente in maniera corretta, in caso positivo il sistema legge i dati e avvia il server normalmente, altrimenti notifica l’anomalia e ripristina il server; 6. Il sistema comunica la sua disponibilità ai client ad offrire servizi; |
| **Condizione di entrata:** | L’amministratore del sistema ha accesso al sistema. |
| **Condizione di uscita:** | Il server è stato avviato. |

|  |  |
| --- | --- |
| **Nome:** | **Spegnimento del server** |
| **Partecipante:** | Amministratore del sito |
| **Flusso degli eventi:** | 1. L’amministratore accede alla macchina in cui è avviato il server; 2. Il tecnico spegne il server; 3. Il sistema comunica ai client la sua terminazione; 4. I client ricevono l’informazione e la registrano; 5. Il sistema mostra la conferma dell’avvenuta operazione; |
| **Condizione di entrata:** | L’amministratore del sistema ha accesso al sistema. |
| **Condizione di uscita:** | Il server è stato spento. |

|  |  |
| --- | --- |
| **Nome:** | **Crash server** |
| **Partecipante:** | Amministratore del sito |
| **Flusso degli eventi:** | 1. La schermata del client visualizza un messaggio di errore che comunica il guasto del sistema e che tutti i servizi di quest’ultimo non saranno disponibili fino a quando il problema tecnico non sarà risolto; 2. L’amministratore provvede al ripristino del Server manualmente; 3. Il Sistema comunica la sua disponibilità ai client ad offrire i propri servizi; |
| **Condizione di entrata:** | Server in crash. |
| **Condizione di uscita:** | Il server è stato ripristinato. |

|  |  |
| --- | --- |
| **Nome:** | **Crash LAN** |
| **Partecipante:** | Amministratore del sito |
| **Flusso degli eventi:** | 1. La schermata del client visualizza un messaggio di errore che comunica il guasto relativo alla LAN e che tutti i servizi di essa non saranno disponibili fino a quando il problema tecnico non sarà risolto; 2. L’amministratore provvede al ripristino della rete LAN manualmente; 3. Il Sistema viene ripristinato; 4. Il Sistema comunica la disponibilità della LAN ai client per offrire i propri servizi; |
| **Condizione di entrata:** | La connessione LAN cade. |
| **Condizione di uscita:** | Il server è stato ripristinato. |

# Servizi sottosistemi

**SS\_Accesso**

|  |  |
| --- | --- |
| Servizio | Descrizione |
| Registrazione | Permette ad un visitatore di effettuare la registrazione al sistema. |
| Login | Permette al cliente di effettuare l’accesso al sistema |
| Logout | Permette al cliente di effettuare il logout dal sistema |

**SS\_GestioneProfilo**

|  |  |
| --- | --- |
| Servizio | Descrizione |
| Modifica | Permette al cliente di effettuare le modifiche al proprio account. |
| Visualizza profilo | Permette al cliente di visualizzare i dati relativi al proprio profilo |
| Aggiunta di un nuovo indirizzo |  |
| Aggiunta di una nuova carta di credito |  |
| Visualizza storico ordini |  |

**SS\_Acquisto**

|  |  |
| --- | --- |
| Servizio | Descrizione |
| Ricerca prodotto |  |
| Visualizza catalogo |  |
| AggiungiProdottoAlCarrello | Permette al cliente di aggiungere un prodotto al carrello |
| EliminaProdottoDalCarrello | Permette al cliente di eliminare un prodotto dal carrello |
| Checkout | Permette al cliente di acquistare i prodotti presenti nel carrello |
| Visualizza carrello |  |

**SS\_GestioneCatalogo**

|  |  |
| --- | --- |
| Servizio | Descrizione |
| AggiungiProdotto | Permette al gestore-prodotti di aggiungere un nuovo prodotto al catalogo |
| ModificaProdotto | Permette al gestore-prodotti di modificare le informazioni relative ad un prodotto presente nel catalogo |
| EliminaProdotto | Permette al gestore-prodotti di eliminare un prodotto dal catalogo |

**SS\_GestioneOfferta**

|  |  |
| --- | --- |
| Servizio | Descrizione |
| AggiungiOfferta | Permette al gestore-offerte di aggiungere una nuova offerta al prodotto |
| ModificaOfferta | Permette al gestore-offerte di modificare le informazioni relative ad un’offerta presente nel sistema |
| EliminaOfferta | Permette al gestore-offerte di eliminare un’offerta |

**SS\_GestioneUtente**

|  |  |
| --- | --- |
| Servizio | Descrizione |
| VisualizzaUtenti | Permette al gestore-utenti di visualizzare gli utenti registrati al sistema |
| EliminaUtenti | Permette al gestore-utenti di eliminare un utente dal sistema |

**SS\_GestioneOrdine**

|  |  |
| --- | --- |
| Servizio | Descrizione |
| ModificaStatoOrdine | Permette al gestore-ordini di modificare lo stato di un ordine |
| Visualizza ordini da spedire e/o terminati | Permette al gestore-ordini di visualizzare gli ordini effettuati dai clienti |
| EliminaOrdine | Permette al gestore-ordini di eliminare un ordine |

**SS\_Amministratore**

|  |  |
| --- | --- |
| Servizio | Descrizione |
| AggiungiGestore | Permette all’amministratore di aggiungere un nuovo gestore |
| RimuoviGestore | Permette all’amministratore di rimuovere un gestore dal sistema |

**SS\_Interfacce**

|  |  |
| --- | --- |
| Servizio | Descrizione |
| Interfacce | sistema che gestisce l’interfaccia grafica |

**SS\_Storage**

|  |  |
| --- | --- |
| Servizio | Descrizione |
| Storage | Permette la gestione dei dati persistenti |

# Glossario

|  |  |
| --- | --- |
| GLOSSARIO |  |
| Sottosistemi | Sistema che fa parte di un sistema più complesso |
| Client | Un qualunque componente che accede ai servizi o alle risorse di un'altra componente detta server |
| Server | Componente proposto a servire servizi ad altri dispositivi detti client |
| JSP | Java Server Page |
| Servlet | Permettono la creazione di applicazioni web |
| HTML | HyperText Markup Language |
| DBMS | Database management system |