

Università degli Studi di Salerno
Corso di Ingegneria del Software

BesTicket
SDD
Versione 1.0

LOGO PROGETTO



Data: 23/12/2017

Coordinatore del progetto:

Nome	Matricola
Andrea De Lucia	
Rita Francese	

Partecipanti:

Nome	Matricola
Myriam Imbriano	0512103618
Manuel Fuschetto	0512103648

Scritto da:	Myriam Imbriano, Manuel Fuschetto
--------------------	-----------------------------------

Revision History

Data	Versione	Descrizione	Autore
23/12/2017	1.0	System Design Document	Myriam Imbriano, Manuel Fuschetto

INDICE

1. Introduzione	4
1.1 Scopo del Sistema	4
1.2 Obiettivi di Design	4
1.3 Definizioni, Acronimi ed Abbreviazioni	5
1.4 Riferimenti	5
1.5 Panoramica	6
2. Architettura Sistema Corrente	7
3. Architettura Sistema Proposto	7
3.1 Panoramica	7
3.2 Decomposizione in sottosistemi	9
3.3 Mapping Hardware/Software	15
3.4 Gestione dati Persistenti	16
3.5 Controllo Accessi e Sicurezza	18
3.6 Controllo Globale del Software	19
3.7 Condizione di Boundary	19
3.7.1 Start-up	19
3.7.2 Terminazione	19
3.7.3 Fallimento	19
4. Servizio dei Sottosistemi	20
4.1 Gestione Artista	20
4.2 Gestione Luogo	20
4.3 Gestione Concerto	20
4.4 Gestione Biglietti	21
4.5 Gestione Utente	21
4.6 Gestione Acquista	22
5. Glossario	23

1. Introduzione

1.1. Scopo del sistema

Al giorno d'oggi le esigenze delle persone sono cambiate rispetto a qualche anno fa, tutti vorrebbero poter comprare qualsiasi cosa in qualsiasi momento non dovendo per forza andare fisicamente in un negozio e attendere il proprio turno, tener conto degli orari di chiusura, oppure pagare il sovrapprezzo che può essere applicato sui biglietti.

Per far fronte a queste esigenze, lo scopo del nostro sistema è quello di realizzare una piattaforma e-commerce, denominata "BesTicket", dove all'utente finale sarà garantito un sito affidabile, veloce e accessibile 24 ore su 24 da qualsiasi dispositivo per l'acquisto online di biglietti per concerti.

1.2. Obiettivi di design

Il sistema dovrà essere efficiente ed intuitivo. Efficiente perché dato un qualsiasi input, il tempo di risposta dovrà essere breve, intuitivo perché l'interfaccia e le funzionalità dovranno essere semplici e alla portata di tutti, così che ogni utente potrà semplicemente sfruttare tutti i servizi del sito.

Alcuni aspetti che avrà il sistema sono:

1.2.1 Criteri di Performance

Tempo di risposta	BesTicket assicura, in breve tempo, una risposta alle richieste dell'utente, la quale verrà elaborata ed inoltrata entro 10 secondi. Il tempo di risposta dipende, ovviamente, dalla velocità della connessione ad Internet, se la connessione è lenta i tempi sopracitati potrebbero non essere rispettati.
Memoria	La quantità di memoria che verrà utilizzata dal sito web non può essere stimata precisamente perché potrebbero essere memorizzati migliaia di utenti, quindi, dovrebbe essere garantita una buona scalabilità.

1.2.2 Criteri di Affidabilità

Robustezza	BesTicket sarà in grado di gestire eventuali input errati senza interrompere il funzionamento dell'intero sistema. Nel caso l'Utente dovesse quindi commettere errori nell'inserimento, il Sistema lo informerà tempestivamente tramite messaggi di errore.
Sicurezza	L'accesso al sito e alle varie funzionalità, disponibili in base al tipo di Utente, è controllato da un sistema di autenticazione che non permette agli utenti non autorizzati di accedere ad alcune aree del Sistema.

1.2.3 Criteri di Manutenzione

Estendibilità	Il sistema dovrà essere propenso all'inserimento di nuove funzionalità. Quindi, si adotterà una programmazione ben strutturata e di facile comprensione.
Modificabilità	Dovrà essere possibile effettuare facilmente modifiche al sistema in caso di bug.

1.2.4 Criteri per l'Utente Finale

Usabilità	BesTicket sarà facile ed intuitivo grazie ad un'interfaccia semplice ed efficace che guiderà l'Utente all'interno del sito.
-----------	---

1.3 Definizioni, Acronimi e Abbreviazioni

BesTicket: Nome del Sistema in Sviluppo;

Utente non Registrato: Utente non Loggato che può usufruire soltanto della ricerca dei concerti;

Utente Registrato: Utente registrato e loggato che può usufruire di tutte le funzioni che la piattaforma mette a disposizione;

Amministratore Concerti: Amministratore registrato e loggato che può usufruire di tutte le funzioni che la piattaforma mette a disposizione;

Amministratore Utenti: Amministratori registrato e loggato che può usufruire di tutte le funzioni che la piattaforma mette a disposizione;

Form: Insieme di una o più Textbox che fornisce all'utente autenticato l'inserimento di determinati requisiti o altro;

JDBC: Java DataBase Connection.

DBMS: Database Management System, Sistema di gestione del database

Mysql: è il più diffuso database Open Source basato sul linguaggio SQL.

RAD: Requirements Analysis Document.

SDD: System Design Documents

1.4 Riferimenti

Object Software Engineering – Using UML, Patterns and Java.
Documento RAD del progetto **BesTicket**.

1.5 Panoramica

È importante fare un accenno alle attività di system design che costituiscono le fondamenta per l'architettura software del sistema.

1. *Decomposizione del sistema*: Il sistema viene suddiviso in diversi sottosistemi.

Ogni sottosistema è caratterizzato da servizi che offre ad altri sottosistemi.

L'insieme dei servizi sarà denominato Interfaccia.

2. *Mapping Hardware/Software*: in questa sezione vengono prese decisioni riguardo le piattaforme hardware scelte per il sistema, la comunicazione tra nodi, e come vengano incapsulati i servizi di un sottosistema.

3. *Gestione dei dati persistenti*: In tale sezione vengono individuati gli oggetti che devono essere resi persistenti e quale software verrà utilizzato per tale scopo.

4. *Politiche di accesso e Sicurezza*: In tale sezione si individueranno grazie ad una tabella le operazioni che ogni attore può effettuare.

5. *Controllo del software globale*: che descrive il modo in cui è implementato il controllo globale del software e come si sincronizzano i sottosistemi.

6. *Condizioni Boundary*: che descriverà oltre l'avvio anche la gestione dei fallimenti, interruzione di corrente o anche a errori di progettazione.

2. Architettura del Sistema Corrente

Attualmente non esiste un sistema software efficiente per la vendita di biglietti per concerti in Italia.

3. Architettura del Sistema proposto

3.1 Panoramica

Il sistema che andremo a sviluppare sarà un e-commerce incentrato sulla vendita di biglietti per concerti.

Ci sono quattro tipologie di utenti i quali possono usare diverse funzionalità e servizi del Sito

1. L'utente registrato deve essere in grado di:

- Effettuare il login.
- Visualizzare la lista dei concerti, relativi agli artisti, in vendita.
- Filtrare i vari concerti per artista, genere e luogo.
- Visualizzare le informazioni personali relative all'account.
- Visualizzare le informazioni relative al sito.
- Cambiare la password.
- Elimina Account.
- Aggiungere biglietti nel carrello o svuotare il carrello.
- Acquistare biglietti.
- Effettuare il log-out.

2. L'utente non registrato deve essere in grado di:

- Effettuare la registrazione
- Visualizzare la lista dei concerti, relativi agli artisti, in vendita.
- Filtrare i vari concerti per artista, genere e luogo.
- Visualizzare le informazioni relative al sito.

3. L'amministratore Concerti deve essere in grado di:

- Effettuare il login.
- Visualizzare la lista dei concerti, relativi agli artisti, in vendita.
- Filtrare i vari concerti per artista, genere e luogo.
- Visualizzare le informazioni personali relative all'account.
- Visualizzare le informazioni relative al sito.
- Cambiare la password.
- Elimina Account.
- Aggiungere un artista.

- Aggiungere un luogo.
- Aggiungere un concerto.
- Aggiungere biglietti.
- Eliminare artista.
- Eliminare luogo.
- Eliminare concerto.
- Eliminare biglietti.
- Effettuare log-out.

4. L'amministratore Utenti deve essere in grado di:

- Effettuare il login.
- Visualizzare la lista dei concerti, relativi agli artisti, in vendita.
- Filtrare i vari concerti per artista, genere e luogo.
- Visualizzare le informazioni personali relative all'account.
- Visualizzare le informazioni relative al sito.
- Cambiare la password.
- Elimina account.
- "Promuovere" un utente registrato ad Amministratore.
- Eliminare Utenti.
- Visualizza informazione base degli Utenti.
- Effettuare il log-out

L'architettura che verrà utilizzata sarà di tipo Client/Server, nello specifico verrà utilizzato il Pattern MVC.

Il server riceverà le richieste da parte del Client e cercherà di rispondere in tempi breve.

Come detto in precedenza utilizzeremo il Pattern MVC.

MVC sta per:

- **Model:** contiene i metodi di accesso ai dati.
- **View:** si occupa di visualizzare i dati all'utente e gestisce l'interazione fra quest'ultimo e l'infrastruttura sottostante.
- **Controller:** riceve i comandi dell'utente attraverso il View e reagisce eseguendo delle operazioni che possono interessare il Model e che portano generalmente ad un cambiamento di stato del View.

L'utilizzo di questo modello comporta numerosi vantaggi, tra cui, la facilità di manutenzione ed estensione

3.2 Decomposizione in Sottosistemi

Il Sistema utilizzerà un'architettura Three-Tier, in particolare:

- **Il Livello di Presentazione:** è il livello composto da tutti i Boundary Object, come le form, che gli utenti vedono e con i quali possono interagire;
- **Il Livello Applicazione:** è il livello composto dagli oggetti responsabili dell'elaborazione dati e di notificare cambiamenti al Livello di Presentazione. Inoltre questo livello comunicherà anche con il database tramite il Livello Database sottostante;
- **Il Livello Database:** è il livello che si occupa dell'immagazzinamento dei dati persistenti e del loro recupero dal database;

Il sottosistema Livello di Presentazione è stato suddiviso in quattro sottosistemi:

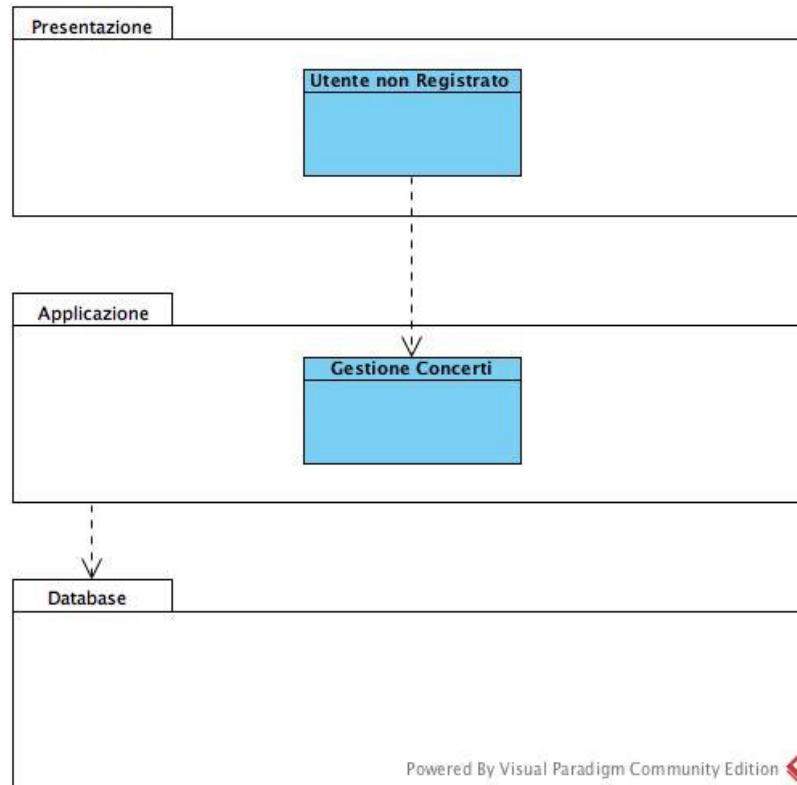
1. **Sottosistema Utente Non Registrato:** include tutte le interfacce grafiche a cui l'Utente non Registrato può accedere ovvero: visualizzare il catalogo, visualizzare informazioni del sito, visualizzare la pagina relativa alla Registrazione e la Homepage;
2. **Sottosistema Utente Registrato:** include tutte le interfacce grafiche a cui l'Utente Registrato può accedere ovvero: visualizzare il catalogo, visualizzare informazioni del sito, visualizzare il carrello e procedere successivamente all'acquisto, visualizzare le proprie informazioni personali e la Homepage;
3. **Sottosistema Amministratore Concerti:** include tutte le interfacce grafiche a cui l'Amministratore Concerti può accedere ovvero: visualizzare il catalogo, visualizzare informazioni del sito, visualizzare le proprie informazioni personali, visualizzare le pagine relative all'amministrazione dei Concerti (es. aggiunta e rimozione di un Concerto o di un Luogo) e la Homepage;
4. **Sottosistema Amministratore Utenti:** include tutte le interfacce grafiche a cui l'Amministratore Utenti può accedere ovvero: visualizzare il catalogo, visualizzare informazioni del sito, visualizzare le proprie informazioni personali, visualizzare le pagine relative all'amministrazione degli Utenti (es. promozione e rimozione di un Utente) e la Homepage;

Il sottosistema Livello di Applicazione è stato suddiviso in vari sottosistemi:

1. **Sottosistema Gestione Concerti:** include tutte le operazioni che permettono l'aggiunta di un Concerto, l'eliminazione di un Concerto, la visualizzazione di un Concerto tramite un filtro e la visualizzazione delle informazioni relative a quel Concerto;
2. **Sottosistema Gestione Biglietti:** include tutte le operazioni che permettono l'aggiunta o l'eliminazione di biglietti per un determinato Concerto;
3. **Sottosistema Gestione Artista:** include tutte le operazioni che permettono l'aggiunta o l'eliminazione di un Artista;
4. **Sottosistema Gestione Luogo:** include tutte le operazioni che permettono l'aggiunta o l'eliminazione di un Luogo;
5. **Sottosistema Gestione Utente:** include tutte le operazioni che permettono la Registrazione di Utente, l'eliminazione di un Utente, promuovere un Utente ad Amministratore, di visualizzare le informazioni relative all'Utente, cambiare le proprie credenziali di Accesso, effettuare il login e il logout;
6. **Sottosistema Gestione Acquista:** include tutte le operazioni che permettono l'aggiunta di biglietti per un concerto nel carrello, svuotare il carrello, visualizzare i biglietti già presenti nel carrello e procedere all'acquisto;

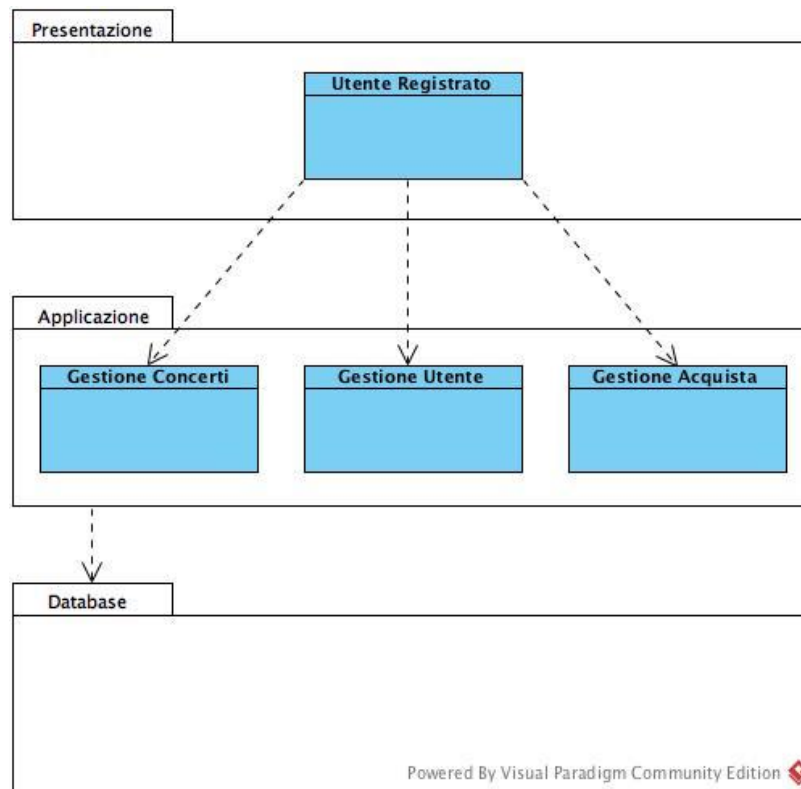
Ora mostriamo nello specifico ogni layer:

Utente non Registrato



Sottosistema Gestione Concerti: include tutte le operazioni che permettono la visualizzazione di un Concerto tramite un filtro e la visualizzazione delle informazioni relative a quel Concerto;

Utente Registrato

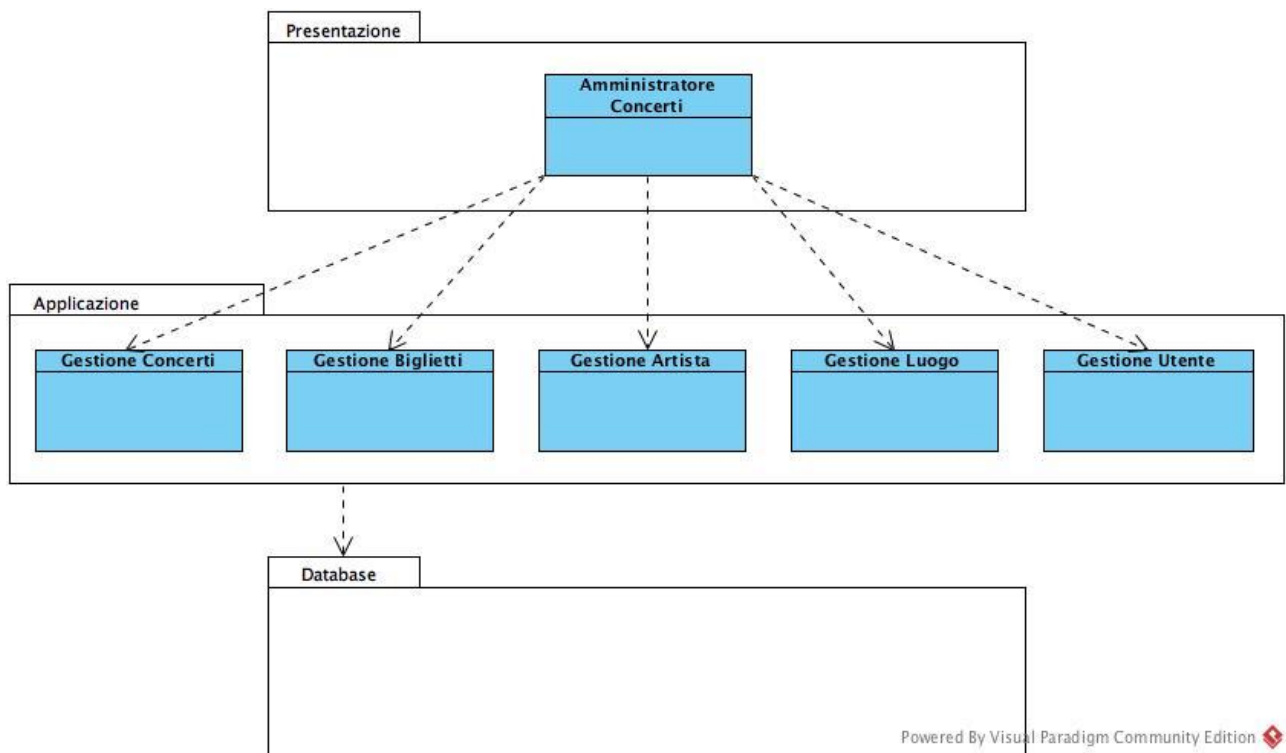


Sottosistema Gestione Concerti: include tutte le operazioni che permettono la visualizzazione di un Concerto tramite un filtro e la visualizzazione delle informazioni relative a quel Concerto;

Sottosistema Gestione Utente: include tutte le operazioni che permettono di visualizzare le informazioni relative all'Utente, cambiare le proprie credenziali di Accesso, effettuare il login e il logout;

Sottosistema Gestione Acquista: include tutte le operazioni che permettono l'aggiunta di biglietti per un concerto nel carrello, svuotare il carrello, visualizzare i biglietti già presenti nel carrello e procedere all'acquisto;

Amministratore Concerti



Sottosistema Gestione Concerti: include tutte le operazioni che permettono l'aggiunta di un Concerto, l'eliminazione di un Concerto, la visualizzazione di un Concerto tramite un filtro e la visualizzazione delle informazioni relative a quel Concerto;

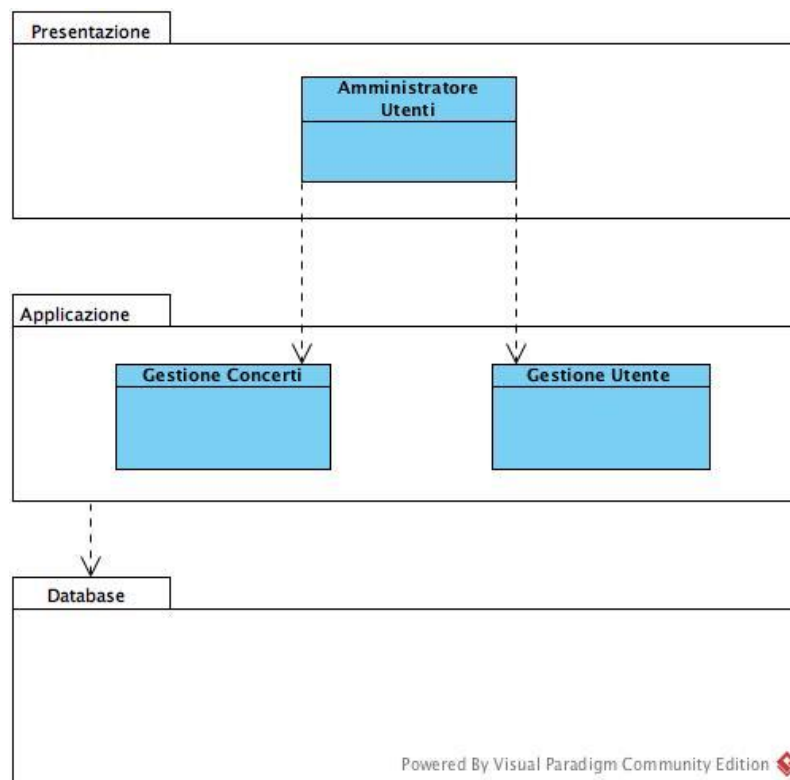
Sottosistema Gestione Biglietti: include tutte le operazioni che permettono l'aggiunta o l'eliminazione di biglietti per un determinato Concerto;

Sottosistema Gestione Artista: include tutte le operazioni che permettono l'aggiunta o l'eliminazione di un Artista;

Sottosistema Gestione Luogo: include tutte le operazioni che permettono l'aggiunta o l'eliminazione di un Luogo;

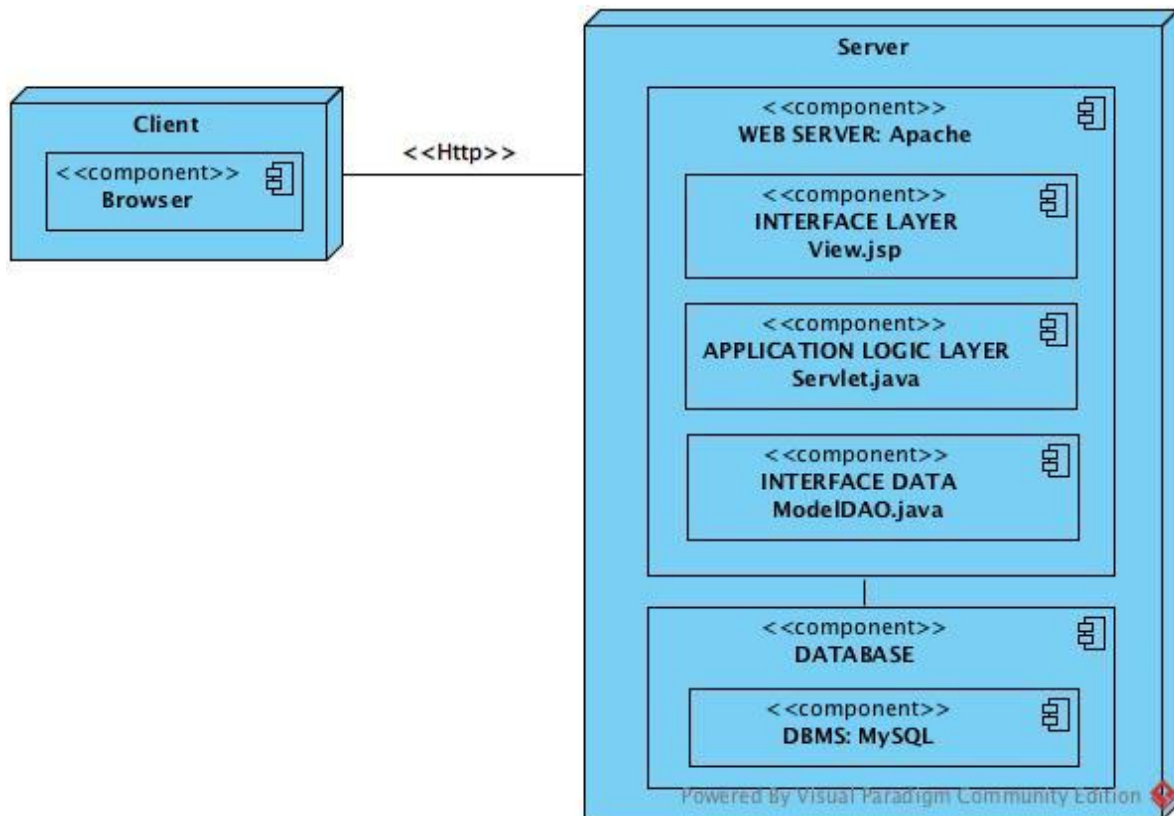
Sottosistema Gestione Utente: include tutte le operazioni che permettono di visualizzare le informazioni relative all'Utente, cambiare le proprie credenziali di Accesso, effettuare il login e il logout;

Amministratore Utenti



7. **Sottosistema Gestione Concerti:** include tutte le operazioni che permettono la visualizzazione di un Concerto tramite un filtro e la visualizzazione delle informazioni relative a quel Concerto;
8. **Sottosistema Gestione Utente:** include tutte le operazioni che permettono l'eliminazione di un Utente, promuovere un Utente ad Amministratore, di visualizzare le informazioni relative all'Utente, cambiare le proprie credenziali di Accesso, effettuare il login e il logout;

3.3 Mapping Hardware/Software



WEB SERVER

Il server utilizzato è Apache Tomcat versione 9.

INTERFACE LAYER

Parti del Sistema contenenti le view che l'utente visualizza tramite Browser.

APPLICATION LOGIC LAYER

Il sistema, e quindi le funzionalità, implementate in linguaggio HTML5 +Java.

INTERFACE DATA

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

Il DBMS usato è MySQL.

3.4 Gestione Dati Persistenti

Artista

NOME	TIPO	VINCOLI	KEY
Cod_Artista	Int	NOT NULL	PRIMARY KEY
Nazionalità	varchar(10)		
Genere	varchar(20)	NOT NULL	
Nome	varchar(50)	NOT NULL	
Tipo	varchar(10)	NOT NULL	
Img	varchar(1000)		
Biglietti_Venduti	Int		

Luogo

NOME	TIPO	VINCOLI	KEY
Cod_Luogo	Int	NOT NULL	PRIMARY KEY
Città	varchar(15)	NOT NULL	
CAP	varchar(5)	NOT NULL	
Via	varchar(100)	NOT NULL	

Concerto

NOME	TIPO	VINCOLI	KEY
Cod_Concerto	Int	NOT NULL	PRIMARY KEY
Data	DATE	NOT NULL	
Ora	TIME	NOT NULL	
Artista	Int	NOT NULL	FOREIGN KEY
Luogo	Int	NOT NULL	FOREIGN KEY

Biglietti

NOME	TIPO	VINCOLI	KEY
Cod_Biglietto	Int	NOT NULL	PRIMARY KEY
Prezzo	Double	NOT NULL	

Utente

NOME	TIPO	VINCOLI	KEY
Username	varchar(50)	NOT NULL	PRIMARY KEY
Nome	varchar(50)	NOT NULL	
Cognome	varchar(50)	NOT NULL	
Via	varchar(50)	NOT NULL	
CAP	char(5)	NOT NULL	
Città	varchar(100)	NOT NULL	
Stato	varchar(25)	NOT NULL	
Tipo	varchar(50)	NOT NULL	
Telefono	varchar(15)		
Pwd	varchar(60)	NOT NULL	
Email	varchar(100)	NOT NULL	
Provincia	varchar(20)		
Data	DATE	NOT NULL	

Acquista

NOME	TIPO	VINCOLI	KEY
Biglietti	int	NOT NULL	PRIMARY KEY, FOREIGN KEY
Utente	varchar(50)	NOT NULL	PRIMARY KEY, FOREIGN KEY
Carrello	tinyint		

3.5 Controllo degli Accessi e Sicurezza

Attore Sottosistema	Utente non Registrato	Utente Registrato	Amministratore Utenti	Amministratore Concerti
Artista				<ul style="list-style-type: none"> ✗ Aggiungere Artista ✗ Eliminare Artista
Concerto	<ul style="list-style-type: none"> ✗ Visualizzazione informazione Concerto ✗ Visualizzazione Concerto tramite Filtro 	<ul style="list-style-type: none"> ✗ Visualizzazione informazione Concerto ✗ Visualizzazione Concerto tramite Filtro 	<ul style="list-style-type: none"> ✗ Visualizzazione informazione Concerto ✗ Visualizzazione Concerto tramite Filtro 	<ul style="list-style-type: none"> ✗ Visualizzazione informazione Concerto ✗ Visualizzazione Concerto tramite Filtro ✗ Aggiungere Concerto ✗ Eliminare Concerto
Luogo				<ul style="list-style-type: none"> ✗ Aggiungere Luogo ✗ Eliminare Luogo
Biglietto				<ul style="list-style-type: none"> ✗ Aggiungere Biglietti ✗ Eliminare Biglietti
Utente	<ul style="list-style-type: none"> ✗ Registrazione 	<ul style="list-style-type: none"> ✗ Login ✗ Logout ✗ Informazioni Personali Account ✗ Cambiare Credenziali Accesso ✗ Eliminare Account 	<ul style="list-style-type: none"> ✗ Login ✗ Logout ✗ Informazioni Personali Account ✗ Cambiare Credenziali Accesso ✗ Eliminare Account ✗ Promuovere Utente ad Amministratore ✗ Eliminare Utente ✗ Visualizzare informazioni base Utente 	<ul style="list-style-type: none"> ✗ Login ✗ Logout ✗ Informazioni Personali Account ✗ Cambiare Credenziali Accesso ✗ Eliminare Account
Acquista		<ul style="list-style-type: none"> ✗ Accedere al Carrello ✗ Svuotare il Carrello ✗ Visualizzare contenuto del Carrello ✗ Procedere all'Ordine 		

3.6 Controllo Globale del Software

Il Sistema “BesTicket” fornisce funzionalità che richiedono una continua interazione da parte dell’Utente, per tale ragione abbiamo un controllo del flusso globale del Sistema di tipo event-driven che permetterà la gestione contemporanea di più Utenti.

3.7 Condizione di Boundary

3.7.1 Start-up

Per lo start-up iniziale del sistema è necessario l’avvio di un web server che fornisca il servizio di un Database MySql per gestire i dati persistenti e l’interpretazione e l’esecuzione del codice dal lato server. Tramite l’interfaccia di Login, sarà possibile autenticarsi tramite credenziali di Accesso (username e password) come utente e, sarà permesso l’accesso a tutte le funzionalità del sistema.

3.7.2 Terminazione

Al momento della chiusura dell’applicazione si ha la terminazione del Sistema con un regolare Logout dal sistema. Viene assicurata la consistenza dei dati, annullando eventuali che erano in esecuzione.

3.7.3 Fallimento

Possono verificarsi diversi casi di fallimento del sistema:

1. Nel caso di guasti dovuti al sovraccarico del database con successivo fallimento dello stesso, è prevista come procedura preventiva il salvataggio periodico dei dati sotto forma di codice SQL per la successiva rigenerazione del DB;
2. Nel caso in cui si verifichi un'interruzione inaspettata dell'alimentazione, non sono previsti metodi che ripristinano lo stato del sistema come prima dello spegnimento inaspettato;
3. Un altro caso di fallimento potrebbe derivare dal software stesso che causa una chiusura inaspettata dovuta ad errori commessi durante la fase di implementazione, non sono previste politiche correttive, l’unico processo che potrà essere eseguito è la chiusura del sistema e il suo successivo riavvio;
4. Un altro caso di fallimento potrebbe essere dovuto ad un errore critico nell'hardware, non è prevista alcuna misura correttiva.

4. Servizi dei Sottosistemi

4.1 Gestione Artista

Sottosistema	Autenticazione
Descrizione	Sottosistema che gestisce tutte le operazioni riguardante l'Artista.
Servizi Offerti	
Servizio	Descrizione
Nuovo Artista	Questa funzionalità permette di aggiungere un Nuovo Artista.
Elimina Artista	Questa funzionalità permette di eliminare un Artista già esistente.

4.2 Gestione Luogo

Sottosistema	Autenticazione
Descrizione	Sottosistema che gestisce tutte le operazioni riguardante il Luogo di un Concerto.
Servizi Offerti	
Servizio	Descrizione
Nuovo Luogo	Questa funzionalità permette di aggiungere un Nuovo Luogo.
Elimina Luogo	Questa funzionalità permette di eliminare un Luogo già esistente.

4.3 Gestione Concerto

Sottosistema	Autenticazione
Descrizione	Sottosistema che gestisce tutte le operazioni riguardante un Concerto.
Servizi Offerti	
Servizio	Descrizione
Nuovo Concerto	Questa funzionalità permette di aggiungere un Nuovo Concerto.
Elimina Concerto	Questa funzionalità permette di eliminare un Concerto già esistente.
Info Concerto	Questa funzionalità permette di visualizzare tutte le informazioni riguardanti un Concerto
Visualizza Concerti per Filtro	Questa funzionalità permette di ricerca all'interno del Sistema Concerti tramite Filtro

4.4 Gestione Biglietti

Sottosistema	Autenticazione
Descrizione	Sottosistema che gestisce tutte le operazioni riguardante i biglietti di un Concerto.
Servizi Offerti	
Servizio	Descrizione
Aggiungi Biglietti	Questa funzionalità permette di aggiungere biglietti riguardanti un Concerto.
Elimina Biglietti	Questa funzionalità permette di eliminare biglietti riguardanti un Concerto.

4.5 Gestione Utente

Sottosistema	Autenticazione
Descrizione	Sottosistema che gestisce tutte le operazioni riguardante l'Utente.
Servizi Offerti	
Servizio	Descrizione
Registrazione	Questa funzionalità permette ad un nuovo Utente di Registrarsi a BesTicket.
Login	Questa funzionalità permette ad un Utente già Registrato di accedere al Sistema con le proprie credenziali per sfruttare tutte le funzionalità disponibili.
Logout	Questa funzionalità permette ad un Utente già Registrato e che ha già effettuato il Login di effettuare il Logout.
Info Utente	Questa funzionalità permette ad un Utente già Registrato e che ha già effettuato il Login di visualizzare le proprie informazioni personali.
Elimina Utente	Questa funzionalità permette ad un Utente già Registrato e che ha già effettuato il Login di eliminare il proprio account. Inoltre, l'Amministratore Utenti sfrutta questa funzionalità per cancellare un Utente dal Sistema.
Cambia Password	Questa funzionalità permette ad un Utente già Registrato e che ha già effettuato il Login di cambiare le proprie credenziali di accesso.
Promuovi Utente	Questa funzionalità permette ad un Amministratore Concerti e che ha già effettuato il Login di promuovere un Utente ad Amministratore.

4.6 Gestione Acquista

Sottosistema		Autenticazione
Descrizione		Sottosistema che gestisce tutte le operazioni riguardante il Carrello e gli Acquisti.
Servizi Offerti		
Servizio		Descrizione
Aggiungi al Carrello		Questa funzionalità permette ad un nuovo Utente di Registrarsi Registrato e che ha già effettuato il Login di aggiungere dei biglietti di un Concerto al Carrello.
Svuota al Carrello		Questa funzionalità permette ad un Utente già Registrato Registrato e che ha già effettuato il Login di svuotare il Carrello.
Carrello		Questa funzionalità permette ad un Utente già Registrato Registrato e che ha già effettuato il Login di visualizzare i prodotti contenuti nel Carrello.
Procedi all'Ordine		Questa funzionalità permette ad un Utente già Registrato Registrato e che ha già effettuato il Login di procedere all'ordine ed acquistare i biglietti relativi a concerti.

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.

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.

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.