

INDICE

- 1. Introduzione**
 - 1.1. Scopo del sistema**
 - 1.2. Design goals**
- 2. Architettura di sistemi affini al software proposto**
- 3. Architettura del software proposto**
 - 3.1. Panoramica**
 - 3.2. Decomposizione in sottosistemi**
 - 3.3. Mapping hardware/software**
 - 3.4. Gestione dati persistenti**
 - 3.5. Controllo degli accessi e della sicurezza**
 - 3.6. Gestione del controllo globale**
 - 3.7. Condizioni boundary**
- 4. Servizi dei sottosistemi**
 - 4.1. SS_01 – Gestione Registrazione**
 - 4.2. SS_02 – Gestione Autenticazione**
 - 4.3. SS_03 – Gestione Account**
 - 4.4. SS_04 – Amministrazione Piattaforma**
 - 4.5. SS_05 – Gestione Eventi**

A cura di

- Cauteruccio Luisa**
- Pascucci Valentina**
- Tortora Carlo**

1.1 Scopo del sistema

1.2 Design goals

Categoria (Performance, Dependability, Cost, Maintenance, End user)	ID	Descrizione Design Goal
<u>Criteri di Prestazioni:</u> Tempo di risposta	DG_01	Il sistema deve garantire la rapidità durante l'esecuzione delle funzionalità a disposizione dell'utente. Tale prestazione sarà influenzata dalla qualità della connessione e dallo stato della batteria del proprio dispositivo durante le pubblicazioni di eventi e proposte di evento.
<u>Criteri di Prestazioni:</u> Throughput	DG_02	Il sistema prevede di gestire un elevato numero di utenze che interagiscono con la piattaforma effettuando pubblicazioni. Il sistema deve garantire rapidità nella comunicazione con il DataBase. Il sistema valuterà la relazione tra le prestazioni e la quantità di risorse impiegate in condizioni prefissate durante l'utilizzo delle sue funzionalità. Il comportamento del sistema nel caso di batteria quasi scarica non garantisce un rapido e corretto funzionamento durante l'interazione da parte dell'utente.
<u>Criteri di Prestazioni:</u> Memoria	DG_03	Il sistema si interfaccia con un DataBase relazionale per la memorizzazione dei dati. La dimensione di tale DataBase dipende dalla quantità di utenze e pubblicazioni effettuati.
<u>Criteri di Affidabilità:</u> Robustezza	DG_04	Il sistema sarà in grado di gestire input errati da parte dell'utente tramite schermate che segnalano lo stato di errore.
<u>Criteri di Affidabilità:</u> Tolleranza all'errore	DG_05	Il sistema sarà in grado di gestire le eccezioni e le situazioni di errore tramite schermate che avvisano l'utente degli eventuali problemi riscontrati e/o che chiedono conferma all'utente nel momento in cui questi effettua una interazione che può essere intesa in diversi modi dal sistema.
<u>Criteri di Affidabilità:</u> Affidabilità	DG_06	Il sistema, dopo una interruzione forzata durante la compilazione di un form da parte di un utente, deve assicurare che i campi già compilati non vadano persi al ripristino della schermata contenente il form.
<u>Criteri di Affidabilità:</u> Sicurezza	DG_07	La sicurezza del sistema è garantita tramite un meccanismo di autenticazione attraverso la schermata del Login che permette il riconoscimento dell'utente tramite la mail di registrazione e una password. La gestione degli utenti e delle pubblicazioni è una delle responsabilità dell'amministratore. Un utente registrato ha accesso ai suoi dati personali, alle proprie pubblicazioni, ma non può gestire i profili e le pubblicazioni altrui.
<u>Criteri di Affidabilità:</u> Disponibilità	DG_08	Il sistema deve essere disponibile 24 ore su 24. L'utente può disporre delle funzionalità del sistema in qualsiasi momento della giornata per interagire con le proprie pubblicazioni e quelle altrui, con i propri dati e profili di ulteriori utenti registrati sulla piattaforma.
<u>Criteri di costo:</u> Sviluppo	DG_09	Il software non richiede costi di sviluppo in quanto le tecnologie utilizzate non prevedono pagamenti per il loro utilizzo. Durante la fase di analisi sono stati utilizzati: Microsoft Office per la documentazione, Visual Paradigm per la rappresentazione visiva di diagrammi, Balsamiq per la realizzazione dei Mockups. Durante la fase di implementazione saranno utilizzati: AndroidStudio per lo sviluppo del codice, MySQL come DBMS relazionale, Eclipse per generare il server, JSON per lo scambio di dati client-server.
<u>Criteri di costo:</u> Installazione	DG_10	Il sistema potrà essere scaricato e installato in modo gratuito dagli utenti interessati che dispongono di un cellulare o tablet che supporta la tecnologia Android, una connessione o traffico dati a internet, un account Google che permetta l'utilizzo del PlayStore.

<u>Criteri di costo:</u> Aggiornamento	DG_11	I vari aggiornamenti del sistema non prevedono costi aggiuntivi.
<u>Criteri di manutenzione:</u> Estensibilità	DG_12	Il sistema prevede la possibilità di aggiungere con facilità nuovi concetti del dominio applicativo dopo la distribuzione del software, quali: definizione di eventi locali in base alla posizione geolocalizzata dell'utente, interazione tramite chat degli utenti, gestione della vendita dei biglietti o prevendite, possibilità di esprimere pareri sulla piattaforma tramite misure di valutazioni personalizzate.
<u>Criteri di costo:</u> Modificabilità	DG_13	Il sistema è sviluppato tramite moduli che permettono modifiche del codice esistente per effettuare correzioni degli errori, aggiornare il codice ed aggiungere nuove funzionalità.
<u>Criteri di costo:</u> Tracciabilità dei requisiti	DG_14	La tracciabilità dei requisiti è garantita dal documento di analisi dei requisiti che permette di mappare quest'ultimi all'interno del codice sviluppato.
<u>Criteri di costo:</u> Leggibilità	DG_15	Il codice sarà organizzato tramite appositi moduli che garantiscono la facilità di interazione con il codice.
<u>Criteri di costo:</u> Adattabilità	DG_16	Il sistema software sviluppato non prevede al momento l'utilizzo su ulteriori domini applicativi, pertanto può essere utilizzato solo su dispositivi che supportano una tecnologia Android.
<u>Criteri di costo:</u> Portabilità	DG_17	Il sistema software sviluppato può essere utilizzato da utenti che dispongono di cellulare o tablet che supportino una tecnologia Android e una connessione a Internet.
<u>Criteri per l'utente finale:</u> Utilità	DG_18	E' previsto l'utilizzo del software da un numero di utenze relativo a: musicisti, promotori di locali e utenti interessate ad ottenere informazioni su eventi locali.
<u>Criteri per l'utente finale:</u> Usabilità	DG_19	Il sistema è sviluppato secondo una modalità user-friendly che permette l'intuitività nell'utilizzo dell'applicazione grazie a schermate e colori che suggeriscono gli scopi delle funzionalità. L'utente finale non avrà bisogno di utilizzare il manuale utente e non dovrà disporre di competenze specifiche riguardanti lo sviluppo e l'utilizzo di applicazioni simili a quella che si intende realizzare, grazie ad una funzionalità interna al software, accessibile dall'utente, che descrive le funzioni del sistema.

2. Architettura dei sistemi affini al software proposto

Il software proposto riscontra delle somiglianze con le seguenti app: ILiveMusic e Vampr, entrambe dedite alle interazioni fra musicisti.

Vampr permette a utenti registrati di ricercare ulteriori utenti e di stabilire un “match” con essi, in modo tale da interagire tramite chat e creare future collaborazioni musicali.

ILiveMusic dà la possibilità a musicisti e dj di cercare nuovi ingaggi, di partecipare ad eventi in corso ricercati tramite appositi filtri. Inoltre, è presente una sezione dedicata all’artista per condividere concerti ed eventi passati in cui si è esibito. Quest’applicazione permette di aumentare la visibilità di un musicista o un dj tramite la condivisione di profili, ingaggi ed eventi.

3. Architettura del sistema proposto

3.1 Panoramica

Il software che si intende realizzare è una applicazione web-based dedicata alle esigenze di musicisti e promotori che intendono interagire musicalmente tramite condivisione di competenze musicali ed eventi. La piattaforma prevede diverse tipologie di utenze: band, singolo musicista, promotore di eventi o semplice spettatore. Una band può cercare un componente, ottenere informazioni su eventi locali e ampliare il pubblico. Un singolo artista gode delle stesse funzionalità della band, può inoltre creare una band tramite ricerche di artisti affini ai propri gusti e generi musicali. Un promotore di un locale, che ha la possibilità di ospitare artisti o band per esibirsi, può creare una nuova inserzione, specificando le varie informazioni sugli spettacoli. Un evento può avere due modalità: la prima consiste nel cercare proposte di band o artisti singoli che vogliono esibirsi, la seconda serve a pubblicizzare la serata e il proprio locale e cercare nuovi spettatori che vogliono partecipare. Uno spettatore è un utente semplice, che scarica l’applicazione per rimanere sempre informato o cercare eventi utilizzando i vari filtri. E’ infine presente la figura di un amministratore, volta a controllare la veridicità delle pubblicazioni e gli utenti iscritti.

3.2 Decomposizione in sottosistemi

Per la realizzazione del software Musicland è stata scelta un’architettura three-layer in versione Client/Server. Questa architettura divide il software in tre livelli, ciascuno dedicato ad un aspetto comunicante con gli altri:

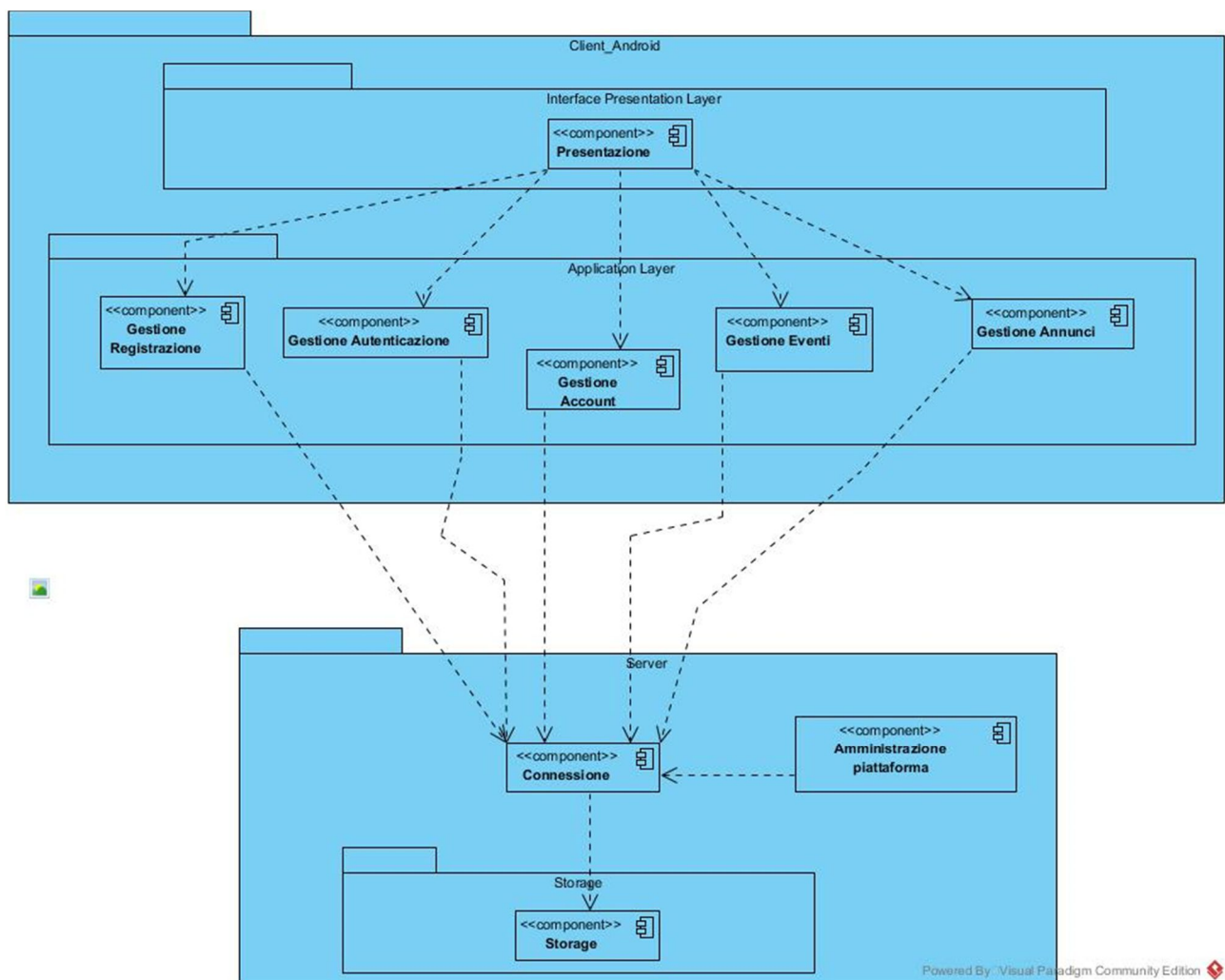
- **Presentation Layer:** include la logica relativa alla presentazione, gestisce quindi l’aspetto grafico dell’applicazione ed è l’unico livello che permette all’utente l’interazione con il sistema.
- **Application Layer:** include la logica di controllo dedita all’elaborazione dei dati provenienti dal livello sovrastante. L’elaborazione delle informazioni avviene tramite la comunicazione con il livello sottostante.
- **Storage Layer:** include la memorizzazione delle informazioni relative ai dati persistenti utilizzando un apposito database al quale l’application layer manda query contenenti richieste e invio di dati.

Il sistema proposto è stato diviso nei seguenti sottosistemi:

- **Gestione Registrazione:** include le operazioni relative alla registrazione dei seguenti utenti: band, artista, promotore, permettendo così a questi ultimi l’utilizzo della piattaforma
- **Gestione Autenticazione:** è dedicato all’ accesso alla piattaforma e all’uscita da quest’ultima tramite rispettivamente le operazioni di login e logout.

- **Gestione Account:** è dedicato agli account di ciascun utente registrato, permette di visualizzare e aggiornare il proprio profilo e le eventuali pubblicazioni, permette la ricerca di ulteriori profili, proposte di evento ed eventi.
- **Amministrazione piattaforma:** fa riferimento alla gestione della piattaforma includendo la figura di un amministratore che garantisca la veridicità degli utenti registrati e le relative pubblicazioni.
- **Gestione Eventi:** è dedicato alla creazione, pubblicazione e aggiornamento di proposte di evento ed eventi da parte di un promotore.

Il seguente diagramma è volto alla rappresentazione dei sottosistemi in cui è stata suddivisa la piattaforma.

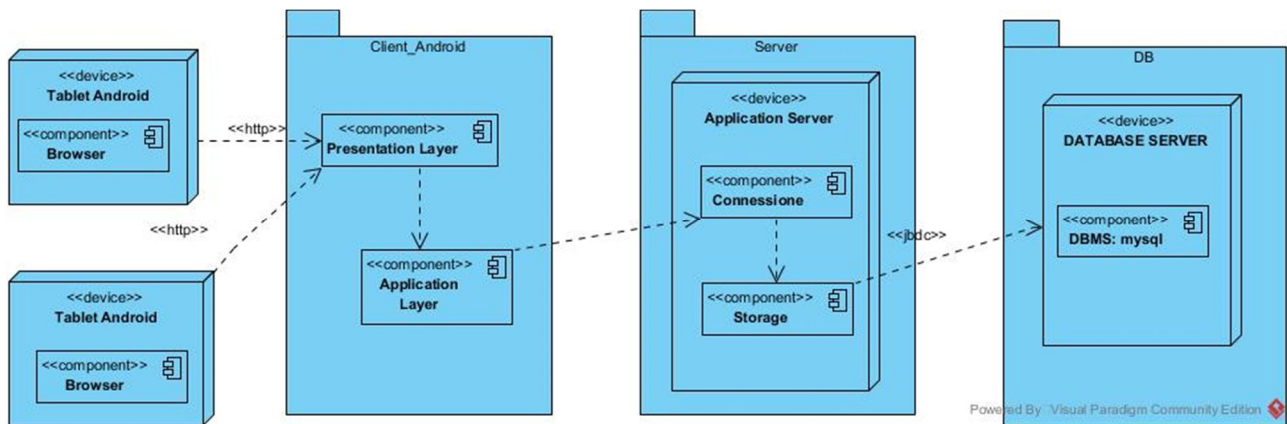


3.3 Mapping hardware/software

La piattaforma che si intende realizzare, per permettere la comunicazione fra i vari sottosistemi, è costituita da un server centrale, diversi client e il database. I client si riferiscono a dispositivi mobili quali telefonini e tablet che supportano una tecnologia Android. Ciascun client effettuerà richieste al server per l'esecuzione

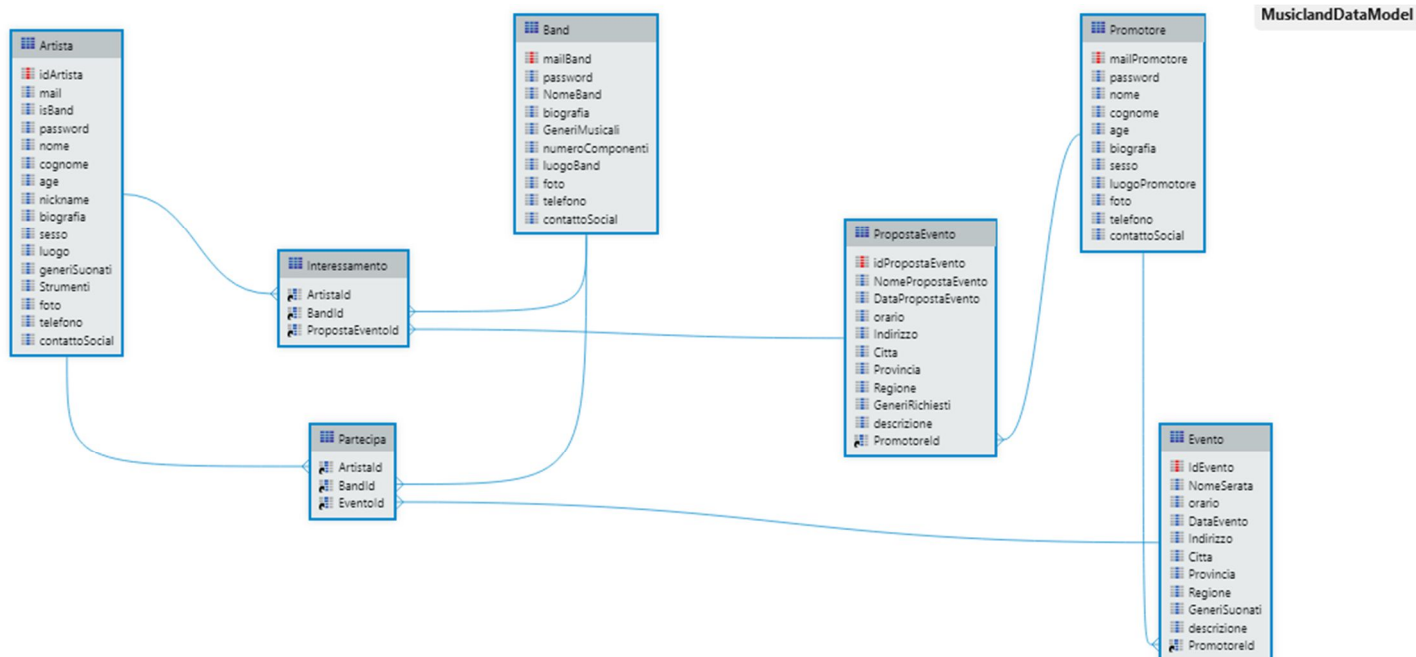
delle funzionalità della piattaforma; il server restituirà ai client le informazioni di cui necessita interfacciandosi con il database che incapsula i dati persistenti relativi agli utenti e alle pubblicazioni. Il Presentation Layer permette all'utente di comunicare con l'applicazione grazie al browser installato all'interno del dispositivo mobile dell'utente. Il linguaggio utilizzato per il livello di presentazione è XML. L'Application Layer, riguardante le funzionalità del sistema, è implementato in linguaggio Java. Lo storage layer è il livello che si interfaccia con il database, occupandosi delle richieste di accesso, inserimento, modifica, eliminazione su dati permanenti presenti all'interno del database.

Di seguito è riportato il diagramma che mostra il mapping sopra descritto.



3.4 Gestione dati persistenti

Diagramma Entity Relationship:



Artista :

idArtista	Mail	isBand	Password	Nome	Cognome	Age

Nickname	Biografia	Sesso	Luogo	GeneriSuonati	Strumenti	Foto	telefono	contattoSocial

Band :

MailBand	Password	NomeBand	Biografia	Luogo	GeneriSuonati	NumeriComponenti	Foto	Telefono	contattoSocial

Promotore :

MailPromotore	Password	Nome	Cognome	Age	Biografia	Sesso	Luogo	Foto	telefono	contattoSocial

Evento :

idEvento	NomeEvento	Data	orario	Indirizzo	Città	Provincia	Regione	GeneriSuonati	Descrizione	Promotoreid

Propostaevento:

idPropostaEvento	NomePropostaEvento	Data	orario	Indirizzo	Città	Provincia	Regione	GeneriRichiesti	Descrizione	Promotoreid

Interessamento :

IdArtista	mailBand	IdPropostaEvento

Partecipazione :

IdArtista	mailBand	IdEvento

Artista:

Nome	Tipo	Null/NotNull	Key
idArtista	Integer(11)	Not null	Key
Mail	Varchar(50)	Not null	
isBand	Boolean	Not null	
Password	Varchar(32)	Not null	
Nome	Varchar(30)	Not null	
Cognome	Varchar(30)	Not null	
Age	integer(10)	Not null	
Nickname	Varchar(30)	Not null	
Biografia	Varchar(1000)	Not null	
Sesso	Varchar(4)	Not null	
Luogo	Varchar(50)	Not null	
GeneriSuonati	Varchar(100)	Not null	
Strumenti	Varchar(50)	Not null	
Foto	Varchar(1000)	Null	
Telefono	Varchar(20)	Not null	
ContattoSocial	Varchar(1000)	Null	

Band:

Nome	Tipo	Null/NotNull	Key
MailBand	Varchar(50)	Not null	Key
Password	Varchar(32)	Not null	
NomeBand	Varchar(50)	Not null	
Biografia	Varchar(1000)	Not null	
Luogo	Varchar(50)	Not null	
GeneriSuonati	Varchar(100)	Not null	
NumeriComponenti	Integer(8)	Not null	
Foto	Varchar(1000)	Null	
Telefono	Varchar(20)	Not null	
ContattoSocial	Varchar(1000)	Null	

Promotore:

Nome	Tipo	Null/NotNull	Key
MailPromotore	Varchar(50)	Not null	Key
Password	Varchar(32)	Not null	
Nome	Varchar(50)	Not null	
Cognome	Varchar(50)	Not null	
Age	integer(5)	Not null	
Biografia	Varchar(1000)	Not null	
Sesso	Varchar(5)	Not null	
Luogo	Varchar(50)	Not null	
Foto	Varchar(1000)	Null	
Telefono	Varchar(20)	Not null	
ContattoSocial	Varchar(1000)	Null	

Evento:

Nome	Tipo	Null/NotNull	Key
idEvento	Integer(11)	Not null	Key
NomeEvento	Varchar(50)	Not null	
Data	Date	Not null	
Orario	Time	Not null	
Indirizzo	Varchar(100)	Not null	
Città	Varchar(100)	Not null	
Provincia	Varchar(100)	Not null	
Regione	Varchar(100)	Not null	
GeneriSuonati	Varchar(100)	Not null	
Descrizione	Varchar(1000)	Not null	
Promotoreid(mailPromotore)	Varchar(50)	Not null	Foreign key

Proposta evento:

Nome	Tipo	Null/NotNull	Key
idPropostaEvento	Integer(11)	Not null	Key
NomePropostaEvento	Varchar(50)	Not null	
Data	Date	Not null	
Orario	Time	Not null	
Indirizzo	Varchar(100)	Not null	
Città	Varchar(100)	Not null	
Provincia	Varchar(100)	Not null	
Regione	Varchar(100)	Not null	
GeneriRichiesti	Varchar(100)	Not null	
Descrizione	Varchar(1000)	Not null	
Promotoreid(mailPromotore)	Varchar(50)	Not null	Foreign key

Interessamento:

Nome	Tipo	Null/NotNull	Key
IdArtista	Integer(11)	NotNull	Foreign key
mailBand	Varchar(50)	NotNull	Foreign key
IdPropostaEvento	Integer(11)	Not Null	Foreign key

Partecipazione:

Nome	Tipo	Null/NotNull	Key
IdArtista	Integer(11)	NotNull	Foreign key
mailBand	Varchar(50)	NotNull	Foreign key
IdEvento	Integer(11)	NotNull	Foreign key

3.5 Controllo degli accessi e della sicurezza

Attori Oggetti	Amministratore	Promotore	Artista	Band	Spettatore
Autenticazione	<ul style="list-style-type: none"> Login Logout 	<ul style="list-style-type: none"> Login Logout 	<ul style="list-style-type: none"> Login Logout 	<ul style="list-style-type: none"> Login Logout 	
Registrazione		<ul style="list-style-type: none"> Registrazione Promotore 	<ul style="list-style-type: none"> Registrazione Artista 	<ul style="list-style-type: none"> Registrazione Band 	
Account		<ul style="list-style-type: none"> Visualizza Profilo Promotore Modifica Profilo Promotore Elimina Profilo Visualizza Mie Proposte Visualizza Mie Eventi Cerca Band & Artisti Cerca Promotore 	<ul style="list-style-type: none"> Visualizza Profilo Artista Modifica Profilo Artista Elimina Profilo Visualizza Mie Eventi Cerca Band & Artisti Cerca Promotore 	<ul style="list-style-type: none"> Visualizza Profilo Band Modifica Profilo Band Elimina Profilo Visualizza Mie Eventi Cerca Band & Artisti Cerca Promotore 	
Eventi		<ul style="list-style-type: none"> Pubblica Proposta Evento Pubblica Evento da proposta Pubblica Evento Modifica Proposta Evento Elimina Propsta Evento Modifica Evento Elimina Evento 	<ul style="list-style-type: none"> Cerca Evento Cerca Proposta Evento 	<ul style="list-style-type: none"> Cerca Evento Cerca Proposta Evento 	<ul style="list-style-type: none"> Cerca Evento
Utenti	<ul style="list-style-type: none"> Visualizza Lista Utenti Elimina Utente 				
Pubblicazioni	<ul style="list-style-type: none"> Visualizza Lista Proposte Evento Visualizza Lista Eventi Elimina Proposta Evento Elimina Evento Visualizza Lista Annunci Elimina Annuncio 				
Annunci			<ul style="list-style-type: none"> Artista Crea Annuncio Artista Modifica Annuncio Artista Cerca Annuncio Elimina Annuncio 	<ul style="list-style-type: none"> Band Crea Annuncio Band Modifica Annuncio Band Cerca Annunci Artisti Elimina Annuncio 	

La piattaforma prevede cinque tipologie di utenze: artista, band, promotore, amministratore, spettatore. Gli accessi sono stati stabiliti in base agli oggetti condivisi tra gli attori operanti nel sistema. Il software prevede che ciascun attore abbia accesso a determinate funzionalità e non ad altre: tale suddivisione è riportata nella sovrastante matrice degli accessi.

3.6 Gestione del controllo globale

Le richieste da parte del client sono accolte dalle classi View che le trasferiscono alle classi Control. Queste ultime gestiscono la logica di controllo comunicando richieste e aggiornamenti al Model. Il server e il client comunicano grazie ad una connessione basata sul protocollo TCP. Tale connessione, e la gestione del server, sono stati gestiti tramite la creazione di una piattaforma dedicata alla parte amministrativa del sistema. Il server è stato creato utilizzando un'architettura Client-Server basata su socket. Tale architettura permette, dopo la creazione del serverSocket, di attivare una specifica funzionalità in grado di attendere la connessione da parte dei client, questo è reso possibile specificando l'ip e la porta del server in attesa. Lo stream dei dati dal client al server sul socket sarà gestito utilizzando le apposite classi che permetteranno di inviare e ricevere stringhe in formato JSON. Il server, tramite classi Control, intercetta ciascuna richiesta del client trasferendola all'opportuna classe Model, che preleva i dati e li restituisce al Control, il quale inoltra, in seguito la risposta al client. Tramite queste interazioni e grazie alla modularità dell'architettura software scelta, il sistema potrà eseguire le proprie funzionalità interagendo anche con i differenti attori. Per la nostra piattaforma si è pensato di utilizzare un meccanismo di controllo di flusso di tipo "event-driven", in quanto la maggior parte delle funzionalità del software saranno scaturite in base al determinarsi di eventi esterni. Tali eventi saranno infatti generati grazie all'interazione degli utenti con la piattaforma, tramite oggetti di tipo boundary che permettono l'interfacciarsi dal punto di vista grafico con il sistema: così, non appena l'utente avrà selezionato un pulsante o una scelta da una lista, sarà generato l'evento associato.

3.7 Condizioni boundary

Le condizioni di boundary fanno riferimento al Server e il Client specifici per l'applicazione. Per quanto riguarda il Server, tali condizioni riguardano l'accensione e arresto del server e la gestione della parte amministrativa del sistema, mentre per il Client gli errori di connessione al server.

Avvio del sistema

Successivamente all'avvio del sistema quest'ultimo mostrerà agli utenti un'interfaccia. Gli utenti autenticati potranno accedere a tutte le funzionalità disponibili per il proprio profilo di registrazione.

Terminazione del sistema

La terminazione del sistema viene eseguita manualmente dall'amministratore, utilizzando l'apposita piattaforma.

Fallimento del sistema

In caso di fallimento del sistema, il Client verrà avvisato tramite schermate che comunicano l'eventuale errore.

Nome scenario	SC_Startup
Partecipanti	Amministratore: Matteo
Flusso di eventi	<ol style="list-style-type: none">1. Matteo, amministratore di Musicland, vuole avviare il sistema e quindi immette le proprie credenziali per avviare il server..2. Il sistema attiva il server e i relativi servizi usufruendo delle apposite procedure di avvio3. Il sistema notifica il successo dell'operazione

Nome scenario	SC_ArrestoServer
Partecipanti	Amministratore: Matteo
Flusso di eventi	<ol style="list-style-type: none"> 1. Matteo vuole arrestare il sistema quindi preme sul pulsante "Arresta". 2. Il sistema controlla eventuali client ancora connessi e ne effettua la disconnessione. Tramite opportune procedure termina i servizi ancora attivi. 3. Il sistema notifica il successo dell'operazione.

4. Servizi dei sottosistemi

SS_01 -Gestione Registrazione

Sottosistema	Gestione Registrazione
Descrizione	Tale sottosistema garantisce tutte le operazioni riguardanti la registrazione degli utenti in tre diversi profili: artista, band, promotore
Servizio	Descrizione
RegistrazioneBand	Questa funzionalità permette a una band di iscriversi alla piattaforma inserendo i propri dati personali, una mail e la password e i campi relativi al proprio profilo
RegistrazioneArtista	Tale funzionalità permette a un artista di iscriversi alla piattaforma inserendo i propri dati personali, una mail e la password e i campi relativi al proprio profilo
RegistrazionePromotore	Questa funzionalità permette a un promotore di registrarsi sulla piattaforma inserendo i propri dati personali, una mail e la password e i campi relativi al proprio profilo

SS_02-Gestione Autenticazione

Sottosistema	Gestione Autenticazione
Descrizione	Questo sottosistema si occupa di tutte le operazioni dedicate all'accesso e alla disconnessione dalla piattaforma da parte degli utenti ed effettua il controllo su i dati inseriti
Servizio	Descrizione
Login	Questa funzionalità garantisce l'accesso all'utente sulla piattaforma inserendo le proprie credenziali: mail e password
Logout	Tale funzionalità permette a un utente di disconnettersi dalla piattaforma
RecuperaPassword	Questa funzionalità permette di recuperare la password di un utente, nel caso in cui quest'ultimo non la ricordi.

SS_03 -Gestione Account

Sottosistema	Gestione Account
Descrizione	Questo sottosistema è dedicato alla gestione dell'account di ogni singolo utente. Comprende operazioni che permettono di visualizzare, modificare ed eliminare il proprio profilo, le proprie proposte di evento e i propri eventi.
Servizio	Descrizione
VisualizzaMioProfilo	Tale funzionalità permette a un utente (indipendentemente dal profilo scelto) di visualizzare il proprio profilo
VisualizzaProfiloArtista	Questa funzionalità permette a un artista di visualizzare il proprio profilo
VisualizzaProfiloBand	Tale funzionalità permette a una band di visualizzare il proprio profilo
VisualizzaProfiloPromotore	Questa funzionalità permette a un promotore di visualizzare il proprio profilo
EliminaProfilo	Questa funzionalità permette ad utente di eliminare il proprio profilo dalla piattaforma
ModificaProfilo	Tale funzionalità permette a un utente(indipendentemente dal profilo scelto) di modificare il proprio profilo
ModificaProfiloArtista	Tale funzionalità permette a un'artista di modificare il proprio profilo
ModificaProfiloBand	Tale funzionalità permette a una band di modificare il proprio profilo
ModificaProfiloPromotore	Tale funzionalità permette a un promotore di modificare il proprio profilo
VisualizzaMieProposte	Questa funzionalità permette a un promotore di visualizzare le proprie proposte di evento.
VisualizzaMieiEventi	Questa funzionalità permette a un promotore di visualizzare gli eventi pubblicati e a una band o artista di visualizzare gli eventi a cui ha partecipato
CercaBand&Artisti	Questa funzionalità permette a band, artisti e promotori di ricercare ulteriori band e artisti utilizzando anche degli appositi filtri
CercaPromotore	Questa funzionalità permette a band, artisti e promotori di ricercare, anche usufruendo di filtri, ulteriori promotori

SS_04 -Amministrazione Piattaforma

Sottosistema	Amministrazione Piattaforma
Descrizione	Tale sottosistema è dedicato alle operazioni svolte dall'amministratore della piattaforma.
Servizio	Descrizione
VisualizzaListaArtisti	Questa funzionalità permette all'amministratore di visualizzare tutti gli artisti iscritti sulla piattaforma
VisualizzaListaBand	Questa funzionalità permette all'amministratore di visualizzare tutte le band iscritte sulla piattaforma
VisualizzaListaPromotori	Questa funzionalità permette all'amministratore di visualizzare tutti i promotori iscritti sulla piattaforma.
EliminaArtista	Questa funzionalità permette all'amministratore di eliminare un artista iscritto sulla piattaforma
EliminaBand	Questa funzionalità permette all'amministratore di eliminare una band iscritta sulla piattaforma
EliminaPromotore	Questa funzionalità permette all'amministratore di eliminare un promotore iscritto sulla piattaforma
VisualizzaListeProposteEvento	Questa funzionalità permette all'amministratore di visualizzare le proposte evento pubblicate dai promotori
EliminaPropostaEventoPromotore	Tale funzionalità permette all'amministratore di eliminare una proposta evento pubblicata in precedenza in caso di contenuti inappropriati
VisualizzaListaEventi	Questa funzionalità permette all'amministratore di visualizzare gli eventi pubblicati dai promotori
EliminaEventoPromotore	Questa funzionalità permette all'amministratore di eliminare un evento pubblicata in precedenza in caso di contenuti inappropriati

SS_05-Gestione Annuncio

Sottosistema	Gestione Annunci
Descrizione	Tale sottosistema è dedicato a tutte le operazioni riguardanti gli annunci, cioè la creazione, eventuali modifiche ed eliminazioni da parte di artisti e band registrati sulla piattaforma
Servizio	Descrizione
BandCreaAnnuncio	Tale funzionalità permette ad una band di creare un annuncio
ArtistaCreaAnnuncio	Questa funzionalità permette ad un artista di modificare un annuncio
BandModificaAnnuncio	Tale funzionalità permette ad una band di modificare un annuncio
ArtistaModificaAnnuncio	Questa funzionalità permette ad un artista di modificare un annuncio
EliminaAnnuncio	Questa funzionalità permette a band e artisti di eliminare un annuncio dalla piattaforma
BandCercaAnnuncio	Tale funzionalità permette ad una band di ricercare, anche utilizzando dei filtri, gli annunci di altri artisti
ArtistaCercaAnnuncio	Tale funzionalità permette ad un artista di ricerca, anche usufruendo di filtri, gli annunci di altri artisti o band.

SS_06 -Gestione Eventi

Sottosistema	Gestione Eventi
Descrizione	Tale sottosistema è dedicato a tutte le operazioni riguardanti eventi e proposte di evento, cioè pubblicazione, modifiche e eliminazioni.
Servizio	Descrizione
CreaPropostaEvento	Tale funzionalità permette a un promotore di creare e pubblicare una proposta di evento
PubblicaEvento(da proposta)	Questa funzionalità permette a un promotore di pubblicare un evento, utilizzando una proposta di evento precedentemente pubblicata
PubblicaEvento(senza proposta)	Tale funzionalità permette a un promotore di pubblicare un evento
ModificaPropostaEvento	Tale funzionalità permette a un promotore di modificare una propria proposta di evento, precedentemente pubblicata
EliminaPropostaEvento	Questa funzionalità permette a un promotore di eliminare una propria proposta di evento
ModificaEvento	Tale funzionalità permette a un promotore di modificare un evento, precedentemente pubblicato
EliminaEvento	Questa funzionalità permette a un promotore di eliminare un evento precedentemente pubblicato
CercaPropostaEvento	Tale funzionalità permette a artisti e band di ricercare proposte di evento
CercaEvento	Questa funzionalità permette a band, artisti e promotori di ricercare eventi.