

TicketStore DB

Elaborato per il corso di Basi di Dati

Mattia Passeri matr. 801497

Indice

1. Analisi dei Requisiti

- 1.1. Intervista
- 1.2. Rilevamento delle ambiguità e correzioni proposte
- 1.3. Definizione delle specifiche in linguaggio naturale ed estrazione dei concetti principali

2. Progettazione Concettuale

- 2.1. Schema scheletro
- 2.2. Raffinamenti proposti
- 2.3. Schema concettuale finale

3. Progettazione Logica

- 3.1. Stima del volume dei dati
- 3.2. Descrizione delle operazioni principali e stima della loro frequenza
- 3.3. Schemi di navigazione e tabelle degli accessi
- 3.4. Raffinamento dello schema
- 3.5. Analisi delle ridondanze
- 3.6. Traduzione di entità e associazioni in relazioni
- 3.7. Schema relazionale finale
- 3.8. Traduzione delle operazioni in query SQL

4. Architettura dell'Applicazione

- 4.1. Architettura dell'Applicazione

1. Analisi dei requisiti

1.1 Intervista

Definizione delle specifiche in linguaggio naturale:

“E’ richiesto lo sviluppo di una base dati che possa essere di supporto al sito web *TicketStore*, che fornisce un servizio di acquisto biglietti per eventi di vario tipo (concerti, spettacoli, mostre...). Dovrà essere consentito creare eventi ed acquistare ticket per tali eventi. Gli utenti devono prima di tutto registrarsi al sito fornendo alcuni dati personali come: nome, cognome, indirizzo (composto da via, numero e città), indirizzo email, e password. Email e password saranno poi utilizzati tutte le volte successive in fase di accesso all’account. Gli utenti possono essere di tre tipi: *bronze*, *silver* oppure *gold*. Al termine della registrazione, ciascun utente può navigare il sito per categorie e scegliere un evento a cui desidera partecipare, visualizzare le informazioni sull’evento ed eventualmente procedere all’acquisto. Per ciascun evento viene visualizzata una pagina web in cui sono riportati il nome, la data, il luogo, il suo organizzatore, una descrizione, un’immagine e le categorie a cui l’evento appartiene. Se decide di procedere all’acquisto, ogni cliente del sito può effettuare un solo ordine per evento e acquistare al massimo 4 biglietti. Per ogni evento sono stampati un certo numero di biglietti, se tutti i biglietti disponibili vengono acquistati, l’evento è sold out e non si può più procedere all’acquisto di alcun biglietto per quel determinato evento. Ogni biglietto è identificato da un numero univoco. Per ciascun evento è possibile che i biglietti appartengano ad una classe, ogni classe ha un nome e il relativo prezzo. L’acquisto di un biglietto da parte di un cliente deve essere memorizzato sulla base dati associando ad esso una data e l’importo pagato; i clienti, dopo aver effettuato l’accesso al loro account, possono visualizzare gli ordini che hanno effettuato e il relativo importo insieme al numero di biglietti acquistati. Gli eventi vengono caricati sul sito dai loro organizzatori, i quali devono essere registrati e per i quali bisogna memorizzare sulla base dati il nome, l’indirizzo (composto da via, numero e città), indirizzo email e password, oltre che una descrizione dell’organizzatore e della tipologia di eventi di cui si occupa. Al momento della creazione di un evento il suo organizzatore deve specificare oltre alle informazioni sull’evento stesso, i biglietti emessi per l’evento e la classe a cui appartengono, non è previsto che possa modificare questa scelta successivamente. Sono presenti anche uno o più amministratori, che si occupano della manutenzione del sito e della base dati. Essi possono effettuare operazioni di cancellazione di utenti, organizzatori ed eventi. Inoltre possono visualizzare tutte le informazioni di interesse relative ad utenti, organizzatori ed eventi. Ciascun nuovo evento, subito dopo la sua creazione, non può essere acquistato finchè non viene approvato da un amministratore. Gli amministratori accedono al sito tramite la propria email e password.”

1.2 Rilevamento delle ambiguità e correzioni proposte

1.2.1 Analisi dei concetti principali e rilevazione ambiguità

In rosso sono sottolineati i concetti principali, in blu i termini che possono essere ambigui perché danno luogo a sinonimie. In marroncino le operazioni principali eseguite sulla base dati.

“E’ richiesto lo sviluppo di una base dati che possa essere di supporto al sito web *TicketStore*, che fornisce un servizio di acquisto biglietti per eventi di vario tipo (concerti, spettacoli, mostre...). Dovrà essere consentito creare eventi ed acquistare ticket per tali eventi. Gli utenti devono prima di tutto registrarsi al sito fornendo alcuni dati personali come: nome, cognome, indirizzo (composto da via, numero e città), email e password. Email e password possono essere utilizzati tutte le volte successive in fase di accesso all’account. Gli utenti possono essere di tre tipi: *bronze*, *silver* oppure *gold*. Al termine della registrazione, ciascun utente può navigare il sito per categorie e scegliere un evento a cui desidera partecipare, visualizzare le informazioni sull’evento ed eventualmente procedere all’acquisto. Per ciascun evento viene visualizzata una pagina web in cui sono riportati il nome, la data, il luogo, il suo organizzatore, una descrizione, un’immagine e le categorie a cui l’evento appartiene. Se decide di procedere all’acquisto, ogni cliente del sito può effettuare un solo ordine per evento e acquistare al massimo 4 biglietti. Per ogni evento sono stampati un certo numero di biglietti, se tutti i biglietti disponibili vengono acquistati, l’evento è sold out e non si può più procedere all’acquisto di alcun biglietto per quel determinato evento. Ogni biglietto è identificato da un numero univoco. Per ciascun evento è possibile che i biglietti appartengano ad una classe, ogni classe ha un nome e il relativo prezzo. L’acquisto di un biglietto da parte di un cliente deve essere memorizzato sulla base dati associando ad esso una data e l’importo pagato; i clienti, dopo aver effettuato l’accesso al loro account, possono visualizzare gli ordini che hanno effettuato e il relativo importo insieme al numero di biglietti acquistati. Gli eventi vengono caricati sul sito dai loro organizzatori, i quali devono essere registrati e per i quali bisogna memorizzare sulla base dati il nome, l’indirizzo (composto da via, numero e città), indirizzo email e password, oltre che una descrizione dell’organizzatore e della tipologia di eventi di cui si occupa. Al momento della creazione di un evento il suo organizzatore deve specificare, oltre alle informazioni sull’evento stesso, i biglietti emessi per l’evento e la classe a cui appartengono, non è previsto che possa modificare questa scelta successivamente. Sono presenti anche uno o più amministratori, che si occupano della manutenzione del sito e della base dati. Essi possono effettuare operazioni di cancellazione di utenti, organizzatori ed eventi. Inoltre possono visualizzare tutte le informazioni di interesse relative ad utenti, organizzatori ed eventi. Ciascun nuovo evento, subito dopo la sua creazione, non può essere acquistato finché non viene approvato da un amministratore. Gli amministratori accedono al sito tramite la propria email e password.”

1.2.2 Glossario dei termini per esplicitare omonimie e sinonimie

Termine	Sinonimo	Descrizione	Collegamenti
Utente	Cliente.	Persona fisica registrata al sito, che effettua acquisti di biglietti di eventi.	Evento, Biglietti.
Evento		Evento organizzato da un organizzatore, per cui i biglietti sono in vendita sul sito.	Organizzatore, Biglietti.
Organizzatore		Azienda che gestisce un evento e mette in vendita biglietti per i suoi eventi attraverso il sito.	Evento, Acquisto.
Amministratore		Gestore del sito, che ha facoltà di apportare modifiche alla base dati.	Organizzatore, Evento.
Biglietto	Ticket.	Biglietto che permette la partecipazione di un utente che lo ha acquistato ad un evento.	Evento, Utente.
Classe	Classe dei biglietti.	Tipologia di biglietto in vendita per un evento, che ha un proprio nome e prezzo.	Biglietto, Ordine, Evento.
Acquisto	Ordine.	Procedura attraverso la quale un utente acquista biglietti per un evento.	Biglietto, Classe, Evento, Utente.

1.3 Definizione delle specifiche in linguaggio naturale ed estrazione dei concetti principali

Frammenti di intervista ristrutturati che mettono in evidenza le specifiche per ogni concetto che interessa modellare.

Utenti:

“Gli utenti devono registrarsi al sito fornendo: nome, cognome, indirizzo, email, e password. Gli utenti possono essere di tre tipi: *bronze*, *silver* oppure *gold*. Gli utenti usano email e password in fase di accesso all’account.”

Eventi:

“Per ciascun evento sono riportati il nome, la data, il luogo, il suo organizzatore, una descrizione, un’immagine, le categorie a cui l’evento appartiene. Per ogni evento sono stampati un numero massimo di biglietti, se tutti i biglietti disponibili vengono acquistati, l’evento è sold out. Ciascun nuovo evento deve essere approvato da un admin prima che gli utenti possano acquistare i biglietti.”

Biglietti:

“Ogni utente può acquistare massimo 4 biglietti per evento. Per ogni evento sono stampati un certo numero di biglietti, se tutti i biglietti disponibili vengono acquistati, l’evento è sold out e non si può più procedere all’acquisto di alcun biglietto per quel determinato evento. Ogni biglietto è identificato da un numero univoco. Per ciascun evento è possibile che i biglietti appartengano ad una classe, ogni classe ha un nome e il relativo prezzo.”

Organizzatori:

“Gli eventi vengono caricati sul sito dai loro organizzatori per i quali bisogna memorizzare il nome, l’indirizzo, città, numero di telefono, indirizzo email e password oltre che una descrizione dell’organizzatore e della tipologia di eventi di cui si occupa.”

Acquisti:

“Al termine della registrazione ciascun utente, se decide di procedere all’acquisto, ogni cliente del sito può effettuare un solo ordine per evento e acquistare al massimo 4 biglietti. L’acquisto di un biglietto da parte di un cliente deve essere memorizzato sulla base dati associando ad esso una data e l’importo pagato; i clienti, dopo aver effettuato l’accesso al loro account, possono visualizzare gli ordini che hanno effettuato e il relativo importo insieme al numero di biglietti acquistati.”

Amministratori:

“Sono presenti uno o più amministratori. Essi possono effettuare operazioni di cancellazione di utenti, organizzatori ed eventi. Inoltre possono visualizzare tutte le informazioni di interesse relative ad utenti, organizzatori ed eventi. Ciascun nuovo evento deve essere approvato da un amministratore. Gli amministratori accedono al sito tramite la propria email e password.”

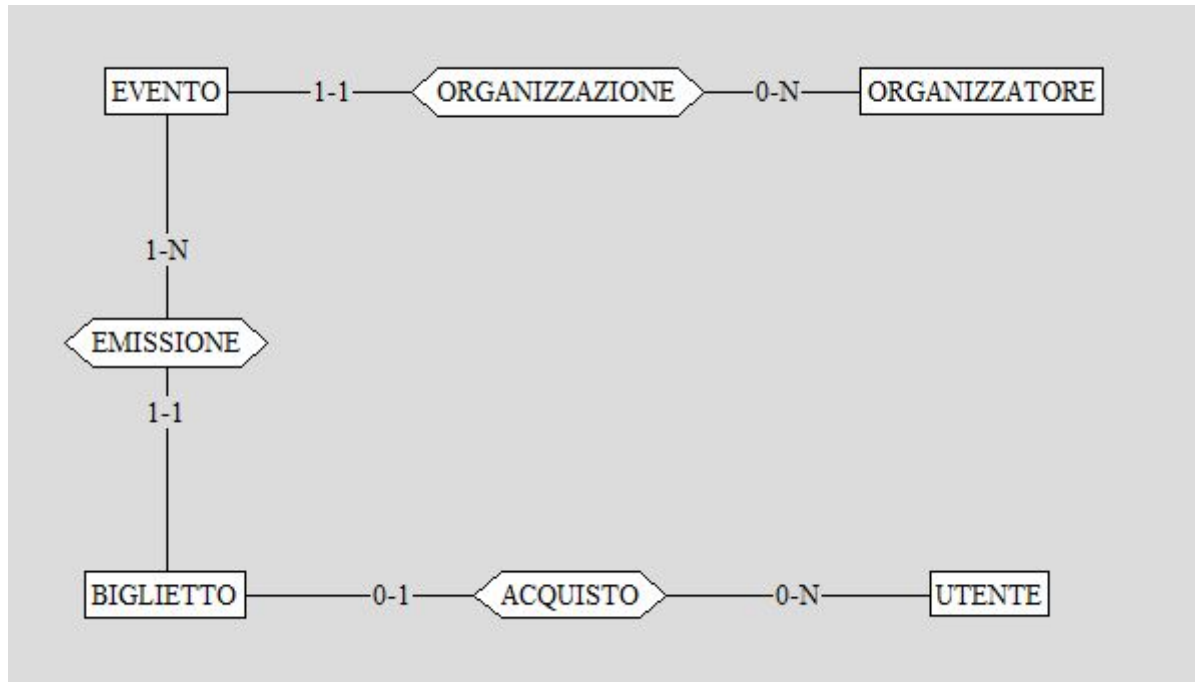
Categorie:

“Al termine della registrazione, ciascun utente può navigare il sito per categorie. Per ciascun evento viene visualizzata una pagina web in cui sono riportate anche le categorie a cui l'evento appartiene.”

2. Progettazione concettuale

2.1 Schema scheletro

Si parte da uno schema portante, in cui sono evidenziate le entità principali per la modellazione della realtà di interesse. Le cardinalità delle associazioni sono indicative in quanto lo schema deve subire ancora diversi passi di raffinamento.

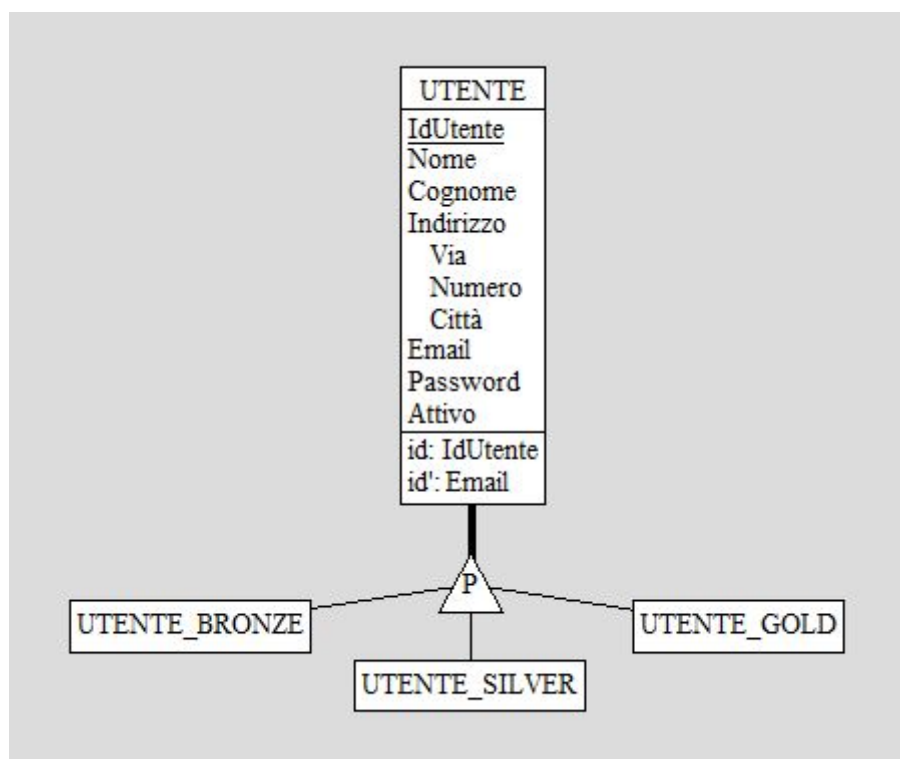


Di seguito i raffinamenti allo schema scheletro che porteranno allo schema concettuale finale.

2.2 Raffinamenti proposti

2.2.1 Utente

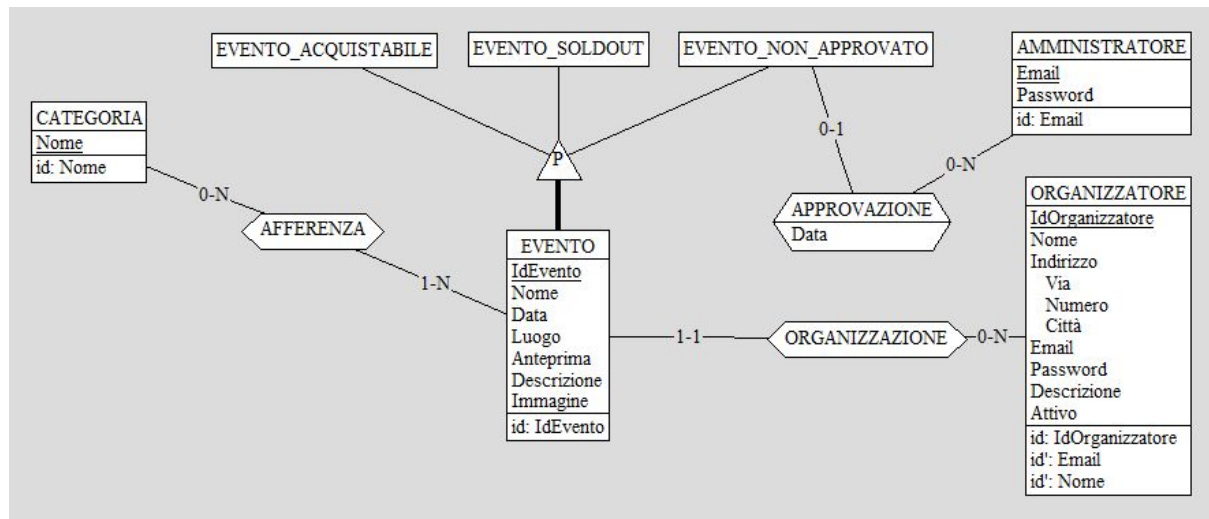
Gli utenti devono registrarsi al sito fornendo: nome, cognome, indirizzo, email, e password, per cui sono predisposti i rispettivi attributi dell'entità UTENTE. Per gli utenti è prevista una generalizzazione totale ed esclusiva che li distingue in: *bronze*, *silver* oppure *gold*. Viene aggiunto un attributo Attivo per modellare l'eliminazione logica degli utenti dal database, senza andare a compromettere le chiavi esterne dell'entità ORDINE che verrà introdotta in seguito. L'attributo Email è chiave di utente in quanto non possono esistere due utenti con la stessa email.



2.2.2 Evento

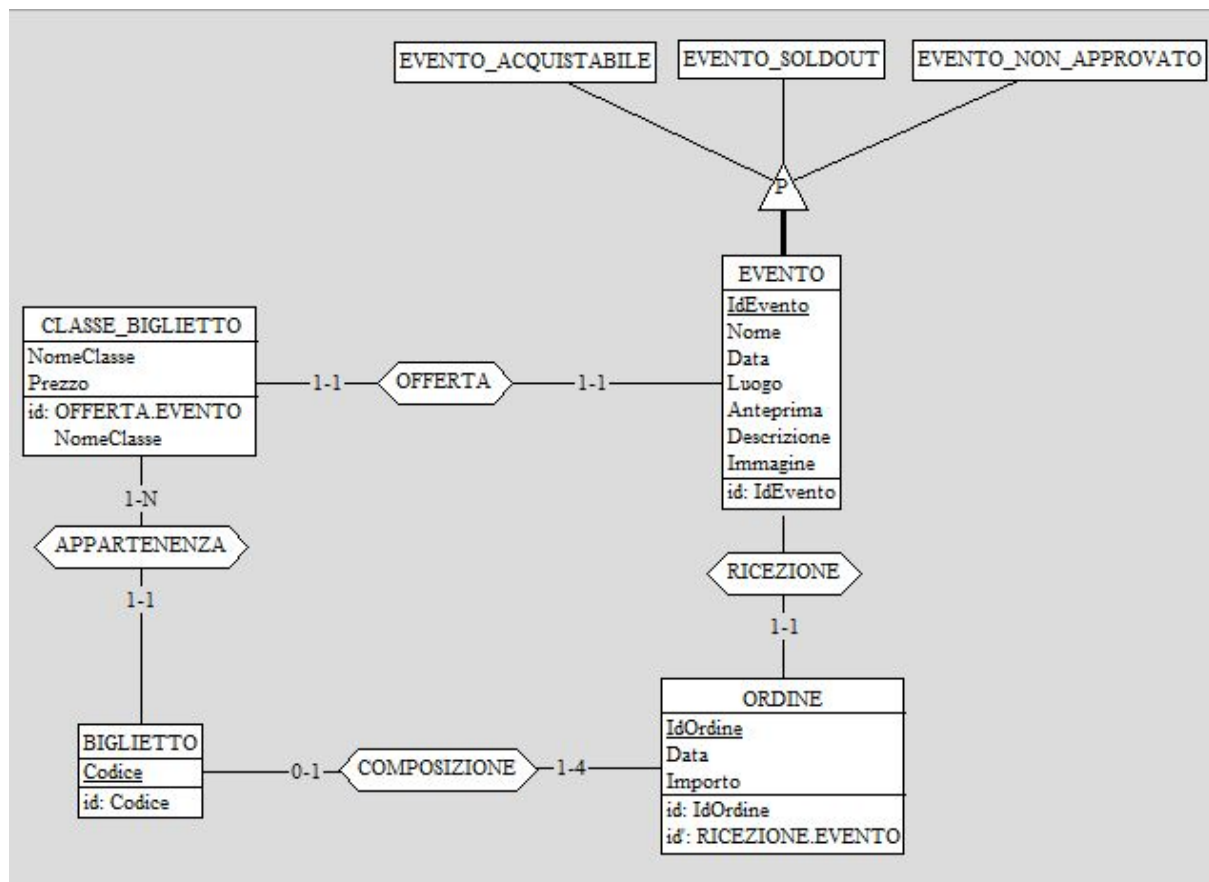
Per ciascun evento devono essere memorizzati nel database: il nome, la data, il luogo, il suo organizzatore, un'anteprima, una descrizione, un'immagine, le categorie a cui l'evento appartiene. Le categorie sono modellate come un'entità collegata agli eventi tramite associazione. Viene aggiunta un'anteprima per ogni evento, ossia un riassunto della sua descrizione, che possa essere utilizzata nella visualizzazione di più eventi in una stessa pagina. L'associazione tra evento ed organizzatore impone che un evento possa appartenere ad un solo organizzatore. Una generalizzazione su evento modella la presenza di eventi non ancora approvati dagli amministratori, eventi acquistabili ed eventi sold out. Si noti che la cardinalità dell'associazione tra Evento e Categoria impone che un evento appartenga ad almeno una categoria.

Questo aspetto permette di ottenere tutti gli eventi per mezzo di ricerche per categoria. Ciascun nuovo evento deve essere approvato da un amministratore.



2.2.3 Biglietto

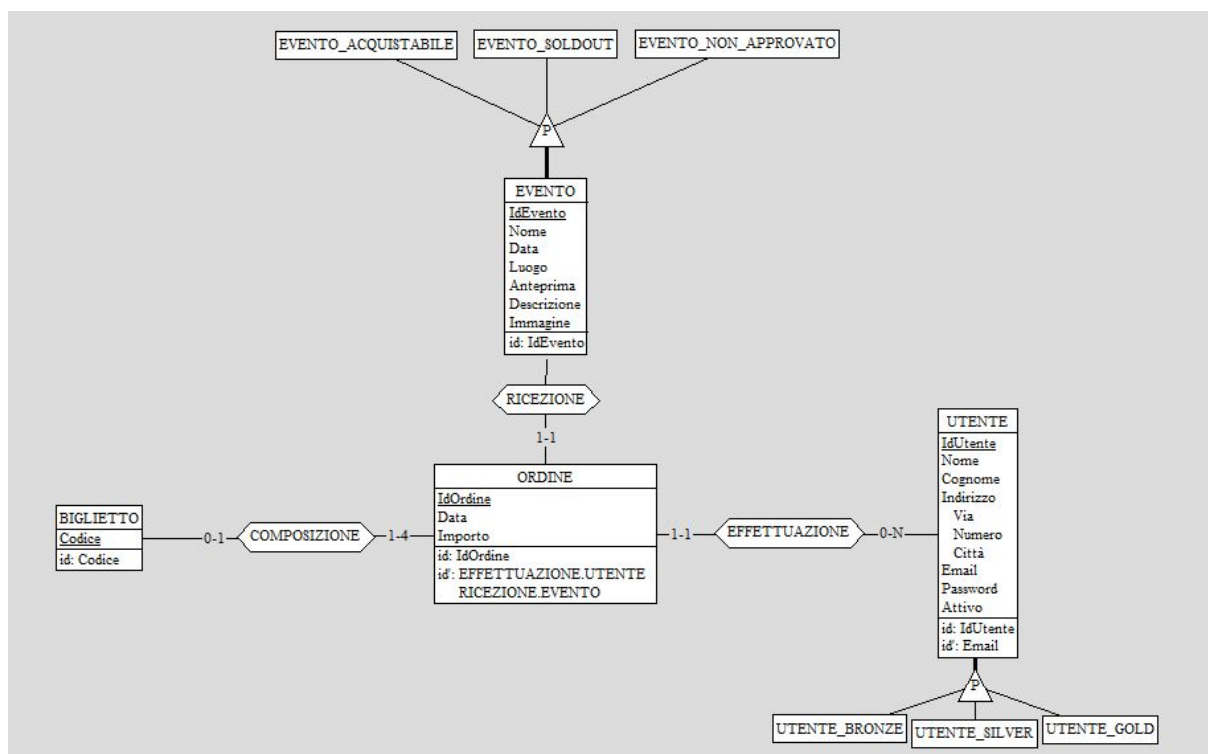
Ogni biglietto è identificato da un numero univoco. Per ciascun evento è possibile che i biglietti appartengano ad una classe, ogni classe ha un nome e il relativo prezzo.



2.2.4 Acquisto

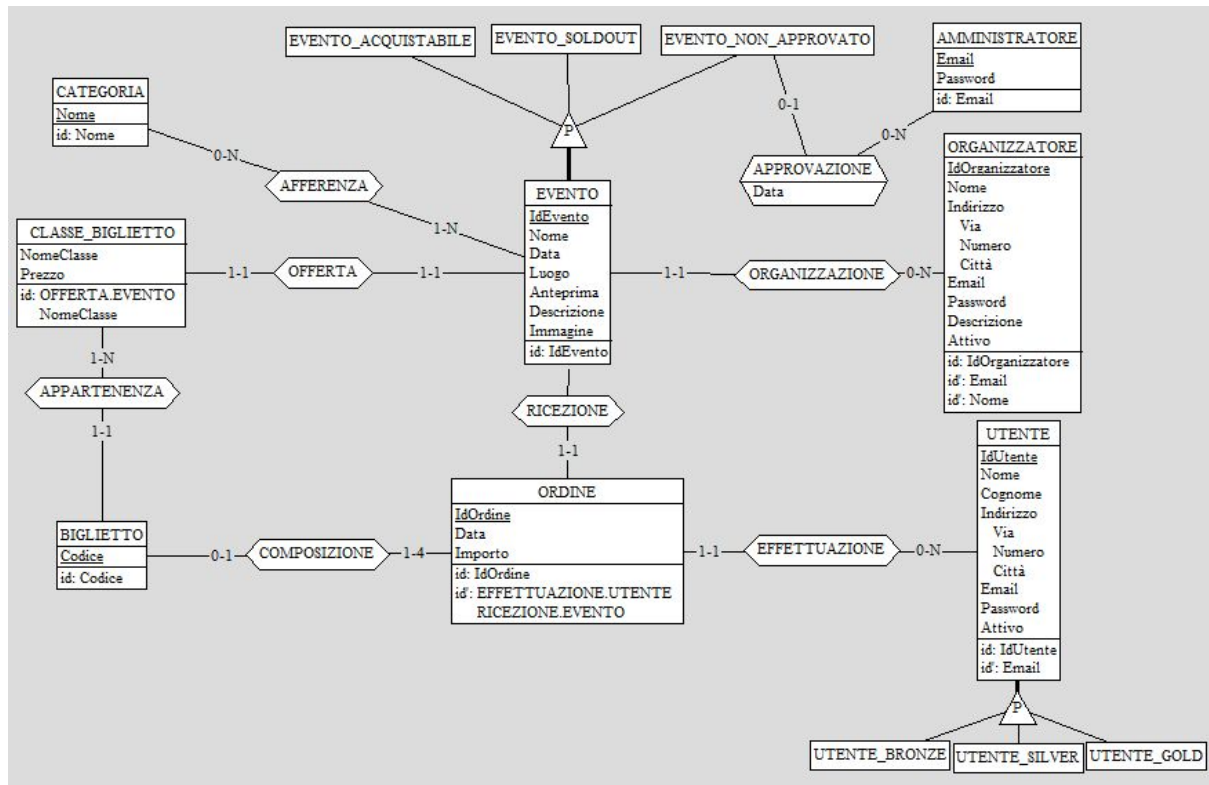
Al termine della registrazione ciascun utente, se decide di procedere all'acquisto, può effettuare un solo ordine per evento ed acquistare al massimo 4 biglietti. L'acquisto di un biglietto da parte di un cliente deve essere memorizzato sulla base dati associando ad esso una data e l'importo pagato. I clienti, dopo aver effettuato l'accesso al loro account, possono visualizzare gli ordini che hanno effettuato e il relativo importo insieme al numero di biglietti acquistati. Per modellare questa specifica con i suoi vincoli, la relazione ACQUISTO dello schema scheletro viene reificata nell'entità ORDINE, che possiede come attributi un codice identificativo, la data dell'ordine e l'importo pagato.

Questa nuova entità partecipa ad associazioni che permettono di far corrispondere ad un ordine: un utente, un evento e da uno a quattro biglietti per l'evento scelto.



2.3 Schema concettuale finale

Per consentire di effettuare operazioni di cancellazione senza effetti in cascata, viene aggiunto un attributo Attivo per Clienti ed Organizzatori. L'attributo Attivo, il cui valore di default è 1 cioè vero, permette di cancellare in maniera non definitiva (logica) clienti ed organizzatori semplicemente impostandolo a 0 cioè falso.



3 Progettazione logica

3.1 Stima del volume dei dati

Per la stima dei volumi dei dati e i successivi calcoli dei costi di accesso si sono ipotizzati valori verosimili per un'implementazione effettiva.

Concetto	Costrutto	Volume
AMMINISTRATORE	E	5
BIGLIETTO	E	70000
CATEGORIA	E	10
CLASSE_BIGLIETTO	E	1000
EVENTO	E	500
ORDINE	E	30000
ORGANIZZATORE	E	100
UTENTE	E	10000
APPROVAZIONE	R	500
APPARTENENZA	R	70000
AFFERENZA	R	2000
COMPOSIZIONE	R	50000
EFFETTUAZIONE	R	30000
OFFERTA	R	1000
RICEZIONE	R	30000
ORGANIZZAZIONE	R	500

3.2 Descrizione delle operazioni principali e stima della loro frequenza

Codice	Utenza	Operazione	Frequenza
1	Utente	Registrazione utente	10 volte al giorno
2	Utente	Ricerca eventi per categorie	5000 volte al giorno
3	Utente	Visualizzazione (utente) evento	1000 volte al giorno
4	Utente	Effettuazione ordine di acquisto	30 volte al giorno
5	Utente	Visualizzazione ordini effettuati e importo pagato	1400 volte al giorno
6	Organizzatore	Registrazione organizzatore	1 volta ogni 10 giorni
7	Organizzatore	Creazione nuovo evento ed emissione biglietti	3 volte a settimana
8	Organizzatore	Modifica evento	1 volta al mese
9	Amministratore	Cancellazione utente	1 volta al mese
10	Amministratore	Cancellazione organizzatore	1 volta all'anno
11	Amministratore	Cancellazione evento	1 volta all'anno
12	Amministratore	Visualizzazione utenti	50 volte al giorno
13	Amministratore	Visualizzazione organizzatori	50 volte al giorno
14	Amministratore	Visualizzazione eventi	50 volte al giorno
15	Amministratore	Approvazione evento	3 volte a settimana

3.3 Schemi di navigazione e tabelle degli accessi.

OP 1) Registrazione utente

Concetto	Costrutto	Accessi	Tipo
UTENTE	E	1	S

Costo totale: $2 \times 10 = 20$ accessi al giorno

OP 2) Ricerca eventi per categorie

Concetto	Costrutto	Accessi	Tipo
EVENTO	E	500	L
AFFERENZA	R	2000	L

Costo totale: $(2000 + 500) \times 5000 = 12500000$ accessi al giorno

OP 3) Visualizzazione (utente) di un evento con categorie

Concetto	Costrutto	Accessi	Tipo
EVENTO	E	1	L
AFFERENZA	R	4	L

Costo totale: $(1 + 4) \times 1000 = 5000$ accessi al giorno

OP 4) Effettuazione ordine di acquisto

Concetto	Costrutto	Accessi	Tipo
CLASSE_BIGLIETTO	E	1	L
ORDINE	E	30002	30001 L e 1 S
COMPOSIZIONE	R	50002	50000 L e 2 S
BIGLIETTO	E	70000	L

Costo totale: $(1 + 30003 + 50004 + 70000) \times 30 = 4500240$ accessi al giorno (per eccesso)

OP 5) Visualizzazione ordini effettuati e importo pagato

Concetto	Costrutto	Accessi	Tipo
EVENTO	E	500	L
ORDINE	E	30000	L
COMPOSIZIONE	R	50000	L

Costo totale: $(50000 + 30000 + 500) \times 1400 = 112700000$ accessi al giorno

OP 6) Registrazione organizzatore

Concetto	Costrutto	Accessi	Tipo
ORGANIZZATORE	E	1	S

Costo totale: $2 \times 40 = 80$ accessi al mese (per eccesso)

OP 7) Creazione nuovo evento ed emissione biglietti

Concetto	Costrutto	Accessi	Tipo
EVENTO	E	1	S
AFFERENZA	R	4	S
CLASSE_BIGLIETTO	R	1	S
BIGLIETTO	E	140	S

Costo totale(limite superiore): $(2 + 8 + 2 + 280) \times 3 = 876$ accessi alla settimana

OP 8) Modifica evento (organizzatore)

Concetto	Costrutto	Accessi	Tipo
EVENTO	E	1	S
AFFERENZA	R	2	S

Costo totale: $(2 + 4) \times 12 = 72$ volte all'anno (media)

OP 9) Cancellazione (logica) utente

Concetto	Costrutto	Accessi	Tipo
UTENTE	E	1	S

Costo totale: $2 \times 12 = 24$ volte all'anno

OP 10) Cancellazione (logica) organizzatore

Concetto	Costrutto	Accessi	Tipo
ORGANIZZATORE	E	1	S

Costo totale: $2 \times 1 = 2$ accessi all'anno

OP 11) Cancellazione evento

Concetto	Costrutto	Accessi	Tipo
AFFERENZA	R	4	S
COMPOSIZIONE	R	100	S
BIGLIETTO	E	280	140L + 140S
ORDINE	E	60	S
CLASSE_BIGLIETTO	E	1	S
APPROVAZIONE	R	1	S
EVENTO	E	1	S

Costo totale: $(8 + 200 + 420 + 120 + 2 + 2 + 2) \times 12 = 9048$ accessi all'anno

OP 12) Visualizzazione utenti

Concetto	Costrutto	Accessi	Tipo
UTENTE	E	10000	L

Costo totale: $10000 \times 50 = 500000$ accessi al giorno

OP 13) Visualizzazione organizzatori

Concetto	Costrutto	Accessi	Tipo
ORGANIZZATORE	E	100	L

Costo totale: $100 \times 50 = 5000$

OP 14) Visualizzazione eventi

Concetto	Costrutto	Accessi	Tipo
EVENTO	E	500	L

Costo totale: $500 \times 50 = 25000$ accessi al giorno

OP 15) Approvazione evento

Concetto	Costrutto	Accessi	Tipo
APPROVAZIONE	R	1	S
EVENTO	E	1	S

Costo totale: $(2 + 2) \times 3 = 12$ accessi a settimana

3.4 Raffinamento dello schema

3.4.1 Eliminazione delle gerarchie

Nello schema E/R sono presenti due gerarchie:

- la prima vede EVENTO come superclasse e le entità figlie rappresentano lo stato in cui si trova un evento, vale a dire se l'evento può essere acquistato perché sono ancora disponibili biglietti, se l'evento è sold out o se l'evento è ancora in attesa di approvazione e quindi i suoi biglietti non sono ancora acquistabili. L'entità EVENTO_NON_APPROVATO partecipa all'associazione APPROVAZIONE con AMMINISTRATORE.

Si è scelto di operare un collasso verso l'alto aggiungendo l'attributo Stato ad EVENTO, in quanto un attributo che indica lo stato dell'evento è più che sufficiente a modellare lo scenario in esame.

Stato assume valore sul dominio {Acquistabile, SoldOut, NonApprovato}.

- la seconda gerarchia differenzia le tipologie di UTENTE in GOLD, SILVER e BRONZE come richiesto dalle specifiche. Anche in questo caso un collasso verso l'alto è più che sufficiente. Viene introdotto l'attributo Tipo in UTENTE, Tipo assume valore nel dominio {Gold, Silver, Bronze}.

Si noti che nell'effettiva implementazione con il DBMS scelto: MySql, si è dovuto implementare le gerarchie come attributi multivalore, in quanto non vi è la disponibilità di una funzione per definire domini utente. A questo proposito sono state introdotte due entità nuove: STATO_EVENTO e TIPO_UTENTE.

3.4.2 Eliminazione attributi composti

Gli unici attributi composti presenti nello schema concettuale sono gli attributi del tutto simili tra loro: Indirizzo, di UTENTE e ORGANIZZATORE. L'attributo Indirizzo viene suddiviso in due parti, una è denominata nuovamente Indirizzo ed è della forma "Via NomeVia Numero", l'altra parte è Città.

3.4.3 Scelta delle chiavi

Le chiavi primarie sono già indicate nello schema concettuale finale. La scelta è ricaduta su l'attributo Nome per CATEGORIA. Sono usati dei codici univoci per identificare UTENTE, ORGANIZZATORE, EVENTO ed ORDINE e BIGLIETTO. La chiave primaria di AMMINISTRATORE è Email. L'entità CLASSE_BIGLIETTO ha un identificatore misto formato dal proprio attributo Nome e da EVENTO.IdEvento, codice dell'evento a cui una classe di biglietti appartiene. La chiave primaria per STATO_EVENTO e TIPO_UTENTE è un identificativo numerico.

3.5 Analisi delle ridondanze

L'unica ridondanza nello schema è la presenza dell'attributo Importo dell'entità ORDINE. Infatti l'importo dovuto per un ordine può essere derivato sommando i prezzi dei biglietti che sono associati all'ordine. Si suppone poi che una volta effettuato un ordine, in un comune scenario operativo della base dati, non sia data la possibilità di apportare modifiche all'ordine stesso. Per questa ragione si può trascurare il costo degli aggiornamenti dei record di ORDINE. Con queste assunzioni conviene sempre mantenere la ridondanza, all'unico costo di poche operazioni per calcolare Importo al momento dell'inserimento di un nuovo record nella relativa tabella.

3.6 Traduzione di entità e associazioni in relazioni

CATEGORIA(Nome)

AMMINISTRATORE(Email, Password)

ORGANIZZATORE(IdOrganizzatore, Nome, Indirizzo, Città, Email, Password, Descrizione, Attivo)

UNIQUE Email, UNIQUE Nome

UTENTE(IdUtente, Tipo, Nome, Cognome, Indirizzo, Città, Email, Password, Attivo)

UNIQUE Email

FK: Tipo REFERENCES Tipo_Utente

AFFERENZA(Nome, IdEvento)

FK: Nome REFERENCES Categoria

FK: IdEvento REFERENCES Evento

APPROVAZIONE(IdEvento, Email)

FK: Email REFERENCES Amministratore

FK: IdEvento REFERENCES Evento

EVENTO(IdEvento, Stato, Nome, Data, Luogo, Antepima, Descrizione, Immagine IdOrganizzatore)

FK: IdOrganizzatore REFERENCES Organizzatore

FK: Stato REFERENCES Stato_Evento (IdStato)

ORDINE(IdOrdine, IdUtente, IdEvento, Data, Importo)

FK: IdUtente REFERENCES Utente

FK: IdEvento REFERENCES Evento

UNIQUE (IdUtente, IdEvento)

BIGLIETTO(Codice, IdEvento, NomeClasse)

FK: (IdEvento, NomeClasse) REFERENCES Classe_Biglietto

CLASSE_BIGLIETTO(NomeClasse, IdEvento, Prezzo)

FK: IdEvento REFERENCES Evento

COMPOSIZIONE(Codice, IdOrdine)

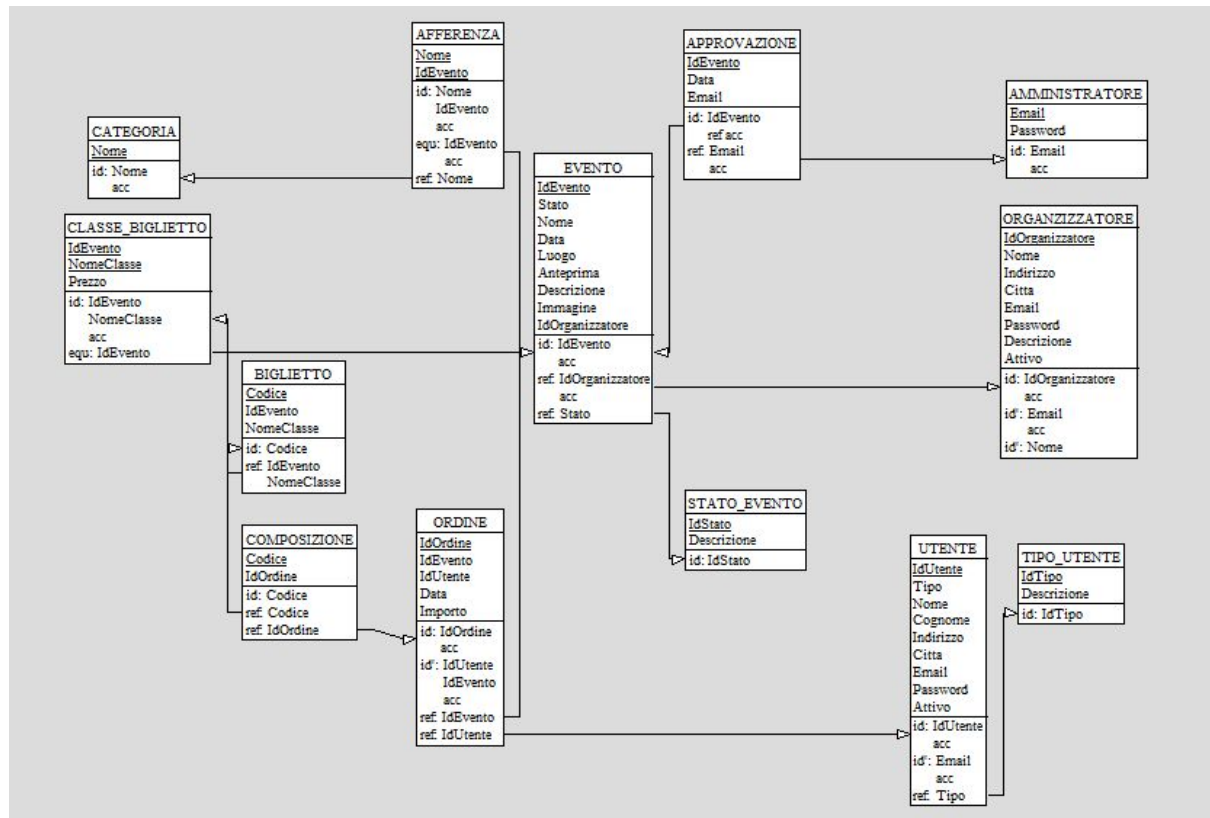
FK: Codice REFERENCES Biglietto

FK: IdOrdine REFERENCES Ordine

STATO_EVENTO(IdStato, Descrizione)

TIPO_UTENTE(IdTipo, Descrizione)

3.7 Schema relazionale finale



3.8 Traduzione delle operazioni in query SQL

OP 1) Registrazione utente

```
INSERT INTO utente (Nome, Cognome, Email, Password, Indirizzo, Citta, Attivo)
VALUES (?, ?, ?, ?, ?, ?, 1)
```

OP 2) Ricerca eventi per categorie

```
SELECT E.IdEvento, E.Nome, E.Data, E.Luogo, E.Immagine, E.Descrizione
E.Anteprima, E.Immagine, E.IdOrganizzatore
FROM evento E, afferenza A
WHERE E.Stato = 1 AND A.Nome=? AND E.IdEvento=A.IdEvento
ORDER BY Data DESC
```

OP 3) Visualizzazione (utente) di un evento con categorie

```
SELECT E.IdEvento, E.Nome, E.Data, E.Luogo, E.Descrizione, E.Anteprima,
E.Immagine, E.IdOrganizzatore,
      (SELECT GROUP_CONCAT(A.Nome)
      FROM afferenza A
      WHERE A.IdEvento=E.IdEvento
      GROUP BY A.IdEvento)
      as categorie
FROM evento E
WHERE E.IdEvento=?
```

OP 4) Effettuazione ordine di acquisto

```
SELECT Prezzo FROM classe_biglietto WHERE IdEvento = ? AND NomeClasse = ?

INSERT INTO ordine (IdEvento, IdUtente, Data, Importo) VALUES (?, ?, ?, ?)

SELECT IdOrdine FROM ordine WHERE IdEvento=? AND IdUtente=?

INSERT INTO composizione (Codice, IdOrdine)
SELECT B.Codice, O.IdOrdine
FROM biglietto B JOIN ordine O ON (B.IdEvento = O.IdEvento)
WHERE (B.Codice) NOT IN (SELECT Codice FROM composizione)
      AND B.IdEvento = ? AND O.IdOrdine = ?

LIMIT ?
```

OP 5) Visualizzazione ordini effettuati e importo pagato

```
SELECT E.IdEvento, E.Nome, E.Luogo, E.Data, E.Descrizione, E.Anteprima,  
E.Immagine, E.IdOrganizzatore, COUNT(*), O.Importo  
FROM evento E JOIN ordine O JOIN composizione C ON (E.IdEvento = O.IdEvento  
AND C.IdOrdine = O.IdOrdine)  
WHERE O.IdUtente=1  
GROUP BY E.IdEvento  
ORDER BY E.IdEvento
```

OP 6) Registrazione organizzatore

```
INSERT INTO organizzatore (Nome, Email, Password, Citta, Indirizzo,  
Descrizione, Attivo) VALUES (?, ?, ?, ?, ?, ?, 1)
```

OP 7) Creazione nuovo evento ed emissione biglietti

```
INSERT INTO evento (Nome, Descrizione, Anteprima, Data, Luogo, Immagine,  
IdOrganizzatore) VALUES (?, ?, ?, ?, ?, ?, ?)  
INSERT INTO afferenza (IdEvento, Nome) VALUES (?, ?)  
INSERT INTO classe_biglietto (IdEvento, NomeClasse, Prezzo, Quantita) VALUES  
(?, ?, ?, ?)  
for ($i = 0; $i < $biglietti; $i = $i + 1) {  
    INSERT INTO biglietto (IdEvento, NomeClasse) VALUES (?, ?)  
}
```

OP 8) Modifica evento

```
UPDATE evento SET Nome = ?, Descrizione = ?, Anteprima = ?, Immagine = ?,  
Luogo = ?, Data = ? WHERE IdEvento = ? AND IdOrganizzatore = ?  
foreach ($categoriedaeliminare as $categoria) {  
    DELETE FROM afferenza WHERE IdEvento = ? AND Nome = ?  
}  
foreach ($categoriedainserire as $categoria) {  
    INSERT INTO afferenza (IdEvento, Nome) VALUES (?, ?)  
}
```

OP 9) Cancellazione utente

```
UPDATE utente SET Attivo = 0 WHERE IdUtente = ? AND Attivo = 1
```


OP 10) Cancellazione organizzatore

```
UPDATE organizzatore SET Attivo = 0 WHERE IdOrganizzatore = ? AND Attivo = 1
```

OP 11) Cancellazione evento

```
DELETE FROM afferenza WHERE IdEvento = ?  
DELETE FROM composizione WHERE Codice IN (SELECT Codice FROM biglietto WHERE  
IdEvento = ?)  
DELETE FROM biglietto WHERE IdEvento = ?  
DELETE FROM ordine WHERE IdEvento = ?  
DELETE FROM classe_biglietto WHERE IdEvento = ?  
DELETE FROM approvazione WHERE IdEvento = ?  
DELETE FROM evento WHERE IdEvento = ?
```

OP 12) Visualizzazione utenti

```
SELECT IdUtente, Email, Nome, Cognome, Citta, Indirizzo  
FROM utente  
WHERE Attivo=1  
GROUP BY Email, Nome  
ORDER BY IdUtente
```

OP 13) Visualizzazione organizzatori

```
SELECT IdOrganizzatore, Email, Nome, Citta, Indirizzo, Descrizione  
FROM organizzatore  
WHERE Attivo=1  
ORDER BY IdOrganizzatore
```

OP 14) Visualizzazione eventi

```
SELECT *  
FROM evento  
ORDER BY IdEvento
```

OP 15) Approvazione evento

```
UPDATE evento SET Stato = 1 WHERE IdEvento = ?  
INSERT INTO approvazione (IdEvento, Email) VALUES (?, ?)
```

4. Architettura dell'Applicazione

4.1 Architettura dell'Applicazione

La base dati progettata è stata impiegata nel contesto specificato nell'intervista, ovvero a supporto di un sito web che offre un servizio di acquisto biglietti online.

L'applicazione che fa uso della base dati è quindi un applicativo web, realizzato con le tecnologie appartenenti allo stack WAMP insieme ad HTML, CSS e Javascript.

Nel sito sono presenti le 3 categorie di utilizzatori individuate nell'intervista: utenti, organizzatori ed amministratori. Le operazioni specifiche per ogni categoria sono disponibili dopo l'accesso ai rispettivi account oppure in seguito ad una nuova registrazione.

L'interfaccia dell'applicativo imita l'interfaccia tradizionale di un sito e-commerce, le interrogazioni alla base dati sono incapsulate in procedure che permettono di mostrare contenuti agli utilizzatori del sito.

Alcune operazioni potrebbero risultare “nascoste” e quindi saranno di seguito illustrate.



Home page del sito con la ricerca per categorie (OP 2) evidenziata in rosso. Da questa schermata tutti i visitatori possono ricercare per categorie gli eventi che sono acquistabili, quindi approvati dagli amministratori e non sold out.

Registrazione Cliente

Nome

Inserisci Nome

Cognome

Inserisci Cognome

Email

Inserisci Email

Password

Inserisci Password

Ripeti Password

Ripeti Password

Indirizzo

Inserisci Indirizzo

Città

Inserisci Città

Torna al Login

Avanti

© Copyright 2020 TicketStore - Tutti i diritti riservati.

[Torna su](#) • [Info](#)

Form di registrazione degli utenti, simile al form di registrazione per gli organizzatori: operazioni (OP 1) e (OP 6).

TICKETSTORE

Accedi

Accesso

Email

Password


Accedi

Non hai un account? Registrati [come Cliente](#) oppure [come Organizzatore](#)

© Copyright 2020 TicketStore - Tutti i diritti riservati.

[Torna su](#) • [Info](#)

Pagina dalla quale effettuare l'accesso al sito con Email e Password per tutte le categorie di utilizzatori.


Giuseppe ▾


Eventi Acquistati

VASCO ROSSI – NON STOP LIVE FESTIVAL

2020-06-26 - Imola

Vasco Non Stop Live Festival: Vasco torna live nel 2020 per 5 concerti nei 4 festival rock più importanti dell'estate!

Biglietti acquistati: 3
Importo: 180 €




Vedi

© Copyright 2020 TicketStore - Tutti i diritti riservati.

[Torna su](#) • [Info](#)

Pagina utente per visualizzazione ordini effettuati e importo pagato (OP 5).


Giuseppe ▾


• Musica • Spettacolo

Imagine Dragons Firenze 2020

2020-06-02 - Firenze

Imagine Dragons in Italia in concerto a Firenze nel 2020 – La band che ha conquistato certificazioni multi-Platino e vinto un Grammy Award, si esibirà live nel nostro Paese il 2 giugno 2020 alla Visarno Arena di Firenze. Disponibile servizio Bus Concerto Imagine Dragons. Quella in Italia sarà l'unica tappa Europea del tour della band capitanata da Dan Reynolds. Sul palco, oltre ai loro più grandi successi, gli Imagine Dragons presenteranno dal vivo i brani contenuti nel quarto album "ORIGINS", "ORIGINS" è stato concepito come un album gemello di "EVOLVE", uscito 16 mesi prima, continuando le esplorazioni sonore del precedente lavoro.

Evento organizzato da:
grandiconcerti@ticketstore.com



Classe Biglietti

© Copyright 2020 TicketStore - Tutti i diritti riservati.

[Torna su](#) • [Info](#)

Pagina di visualizzazione evento per utenti, con i pulsanti per effettuare l'acquisto di biglietti per l'evento (max. 4 biglietti), operazioni: (OP 3) e (OP 4).

Inserisci Evento

Nome Evento:

Data Evento: Luogo Evento:

Descrizione Evento:

Anteprima Evento:

Biglietti

Classi: Classi Biglietti: Quantità: Biglietti Disponibili: Prezzo:

Immagine Evento: [Scegli file](#) Nessun file selezionato

Categorie:

Cinema Cinema OC Event Cinema OC Musica Dibattito

[Inserisci Evento](#) [Annulla](#)

© Copyright 2020 TicketStore - Tutti i diritti riservati.
[Torna su](#) [Info](#)

Form per la creazione e modifica di un evento da parte di un organizzatore (OP 7) e (OP 8).

Clienti

Admin: Admin

- Eventi
- Organizzatori
- Clienti
- Approva Eventi
- Esci

ID: 1

- Nome: Giuseppe
- Cognome: Zucchi
- Email: giuseppez@ticketstore.com
- Indirizzo: Viale Elai 69
- Città: Cesenatico

[Cancella](#)

ID: 2

- Nome: Mariella
- Cognome: Pecoraro
- Email: mariellapecoraro@ticketstore.com
- Indirizzo: Via Vai 2
- Città: Taormina

[Cancella](#)

ID: 3

- Nome: Luca
- Cognome: Bianchi
- Email: luca.bianchi@ticketstore.com
- Indirizzo: Largo Lago 7
- Città: Rimini

[Cancella](#)

ID: 4

- Nome: Pinco
- Cognome: Pallino
- Email: pallino@ticketstore.com
- Indirizzo: Via Rossi 24
- Città: Forlì

[Cancella](#)

ID: 5

- Nome: Floyd
- Cognome: Mayweather
- Email: f.mayweather@ticketstore.com
- Indirizzo: Route 66
- Città: New York

[Cancella](#)

© Copyright 2020 TicketStore - Tutti i diritti riservati.
[Torna su](#) [Info](#)

Pagina di visualizzazione degli utenti del sito con opzione per effettuare la cancellazione di un utente alla volta. Questa pagina incapsula le operazioni: (OP 12) e (OP 9). Dal menu a tendina si possono visitare altre tre pagine simili che forniscono l'interfaccia per tutte le operazioni dell'amministratore.