

ConcertONE : progetto per il corso di Basi di Dati

Bertoni Eleonora

8 Gennaio 2021

Indice

1	Analisi dei requisiti	2
1.1	Intervista	2
1.2	Rilevamento delle ambiguità	3
1.3	Definizione delle specifiche in linguaggio naturale ed estrazione dei concetti principali	4
2	Progettazione Concettuale	5
2.1	Progettazione dello schema E/R	5
2.2	Schema concettuale finale	11
3	Progettazione logica	12
3.1	Stima del volume dei dati	12
3.2	Descrizione delle operazioni principali e stima della loro frequenza	13
3.3	Schemi di navigazione e tabelle degli accessi	14
3.4	Analisi delle ridondanze	29
3.5	Raffinamento dello schema	30
3.5.1	Eliminazione delle gerarchie	30
3.5.2	Scelta delle chiavi primarie	30
3.6	Schema concettuale con aggiunta di ridondanza	31
3.7	Traduzione di entità e associazioni in relazioni	31
3.8	Schema logico finale	34
3.9	Traduzione delle operazioni in query SQL	34
4	Progettazione dell'applicazione	40
4.1	Descrizione dell'architettura dell'applicazione	40

Capitolo 1

Analisi dei requisiti

L'obiettivo è realizzare un database per una società che si occupa dell'organizzazione di concerti e della vendita di biglietti ad essi legata.

Attraverso l'applicativo finale sarà possibile, per gli utenti, cercare informazioni sui concerti (anche passati) e comprare biglietti.

Per gli amministratori saranno disponibili funzioni per l'aggiornamento della base dei dati nonché di reperimento di informazioni relative agli incassi.

1.1 Intervista

La società BigliettONE, che si occupa dell'organizzazione di concerti sul territorio, ha richiesto una base di dati a supporto di un sito in cui verranno caricate le informazioni relative agli eventi. Il sito vuole inoltre gestire un canale per l'acquisto dei biglietti.

Di ogni artista (cantante, dj, musicista...), registrato univocamente tramite codice identificativo, si vogliono salvare il nome, il cognome, il nome d'arte, l'anno di inizio attività e l'anno di fine attività se presente. Per i musicisti va aggiunto lo strumento musicale. Un artista quando organizza un tour o registra\canta un brano lo può fare come solista, come parte di una band oppure facendo una collaborazione con altri artisti.

Per ogni tour organizzato si desidera memorizzare l'artista che lo tiene, il titolo, una breve descrizione e le tappe stabilite.

Per ciascuna tappa si specificano la data, l'ora, la città e il luogo in cui si tiene. Si vuole inoltre conoscere la scaletta delle esibizioni: è importante tener conto dell'ordine delle esibizioni, del brano eseguito e degli artisti che

saliranno sul palco. Se desiderato sarà possibile caricare il video di alcune delle esibizioni di un concerto. Per ogni brano si registrano il titolo, il testo se presente, l'anno di pubblicazione, l'artista originale e gli artisti di cui si vogliono menzionare variazioni (cover/remix) del pezzo. Per ogni location va indicata la tipologia (palazzetto, spazio all'aperto, discoteca...), l'indirizzo, la capienza massima e se presenta posti numerati o in piedi. Se prevista va indicata la suddivisione dei posti in settori.

Il sito deve permettere ad un utente di registrarsi salvando alcuni dati quali nome, cognome e mail e opzionalmente numero di telefono. Il login avviene con l'inserimento della mail e della password. L'utente potrà acquistare biglietti per i concerti. Il prezzo di un biglietto è determinato dal concerto e dall'appartenenza di un posto ad un determinato settore.

1.2 Rilevamento delle ambiguità

Termine	Descrizione	Sinonimi
Artista	Colui che ha una carriera nell'ambito musicale	-
Tour	La tipologia di evento che consideriamo, solitamente prevede più tappe	-
Tappa	Indica l'effettivo svolgimento di un concerto in data e location stabiliti	Concerto
Scaletta	Insieme di esibizioni previste per un determinato concerto	-
Esibizione	Indica l'esecuzione di un determinato brano da parte di uno o più artisti	-
Variazione	Modifica di un brano di cui si vuole tenere traccia	remix, cover (in questo contesto)
Location	Indica dove si svolge un concerto	luogo
Settore	Sezione in cui sono suddivisi i posti di una location adibita ad un concerto. Si può trattare di una suddivisione legata al prezzo dei biglietti, dovuta alla struttura della location o ad entrambe	-
Utente	Colui che accede al sito	-

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

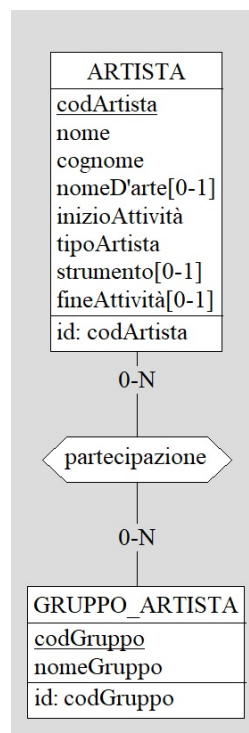
Di ogni **artista** (cantante, dj, musicista...), registrato univocamente tramite codice identificativo, si vogliono salvare il nome, il cognome, il nome d'arte, l'anno di inizio attività e l'anno di fine attività se presente. Per i musicisti va aggiunto lo strumento musicale. Un artista quando organizza un tour o registra\canta un brano lo può fare come **solista**, come parte di una **band** oppure facendo una **collaborazione** con altri artisti. Per ogni **tour** organizzato si desidera memorizzare l'artista che lo tiene, il titolo, una breve descrizione e le tappe stabilite. Per ciascuna **tappa** si specificano la data, l'ora, la città e la **location** in cui si tiene. Si vuole inoltre conoscere la scaletta delle **esibizioni**: è importante tener conto dell'ordine delle esibizioni, del **brano** eseguito e degli artisti che saliranno sul palco. Se desiderato sarà possibile caricare il video di alcune delle esibizioni di un concerto. Per ogni brano si registrano il titolo, il testo se presente, l'anno di pubblicazione, l'artista originale e gli artisti di cui si vogliono menzionare **variazioni** (cover/remix) del pezzo. Per ogni location va indicata la tipologia (palazzetto, spazio all'aperto, discoteca...), l'indirizzo, la capienza massima e se presenta posti numerati o in piedi. Se prevista va indicata la suddivisione dei posti in **settori**. Il sito deve permettere ad un **utente** di registrarsi salvando, tramite nome utente univoco, alcuni dati quali nome, cognome e mail e opzionalmente numero di telefono. Il login avviene con l'inserimento della mail e della password. L'utente potrà acquistare **biglietti** per i concerti. Il prezzo di un biglietto è determinato dal concerto e dall'appartenenza di un posto ad un determinato settore.

Capitolo 2

Progettazione Concettuale

2.1 Progettazione dello schema E/R

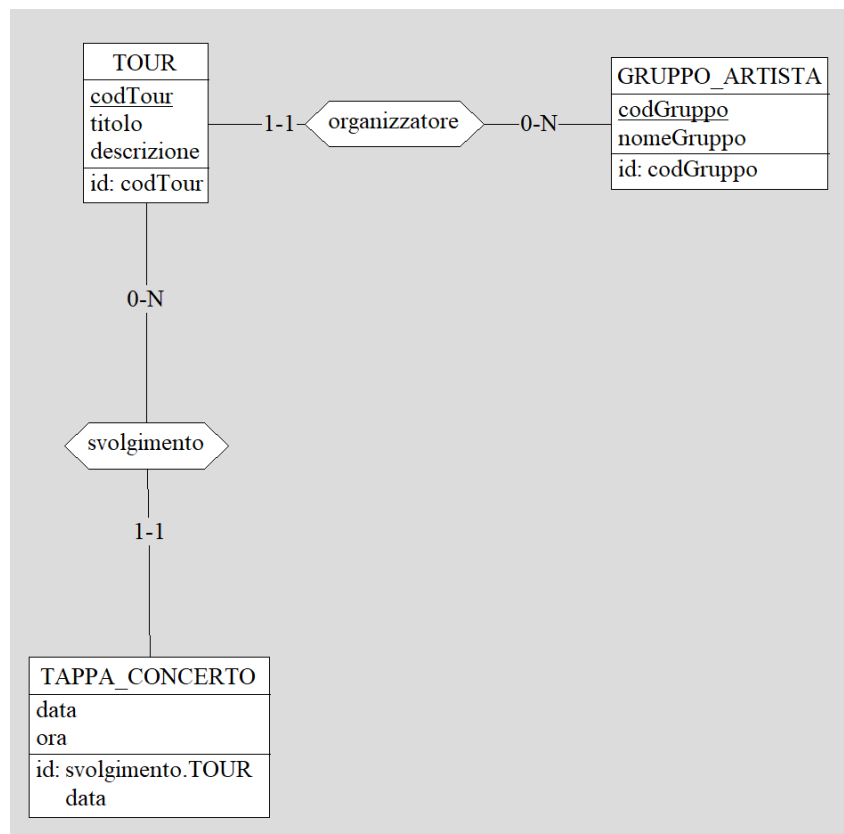
Sviluppo dell'ambito "Artista - Gruppo Artista"



In questo ambito viene modellato il concetto di ARTISTA. Dal momento che sono indicate alcune tipologie di artista e l'unica specifica è costituita

dall'inserimento dello strumento per i musicisti, invece di ricorrere a una gerarchia si è pensato direttamente di introdurre nell'entità un attributo tipo e un attributo opzionale per lo strumento. Si ipotizza quindi che sia utile solo memorizzare lo strumento principale di un musicista. Dalle specifiche emerge che sia centrale un'entità che possa modellare un solista, un gruppo o una collaborazione di artisti che cantano un brano o organizzano un tour. A tale scopo è stata introdotta l'entità GRUPPO_ARTISTA mediante la quale è facilmente indicare il nome del gruppo e, attraverso l'associazione partecipazione, gli artisti che ne fanno parte, da 1 a n elementi.

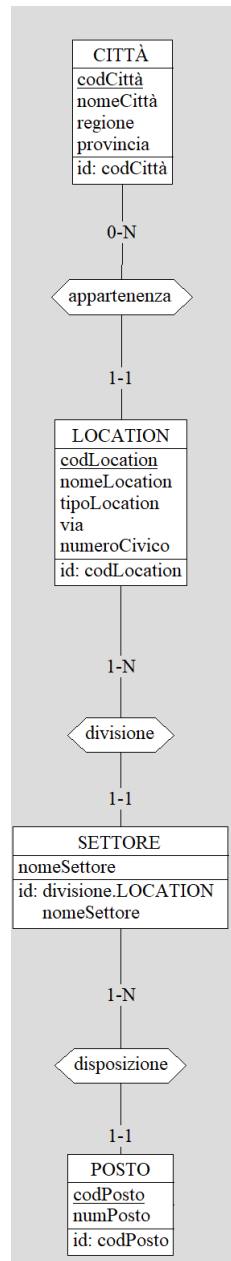
Sviluppo dell'ambito "Tour-Tappa Concerto"



Dalle specifiche si deduce che un GRUPPO_ARTISTA può organizzare dei TOUR ognuno dei quali si suddivide in tappe. Una tappa rappresenta lo svolgimento vero e proprio del concerto ed è modellato con l'identità TAPPA_CONCERTO. Si suppone che un gruppo non possa fare più di un concerto

al giorno per cui le istanze di TAPPA_CONCERTO sono univocamente identificate dall'appartenenza al tour e dalla data della tappa. Infine si considera che un gruppo non possa intraprendere più tour nello stesso periodo, questo aspetto non può essere modellato in E/R e verrà verificato.

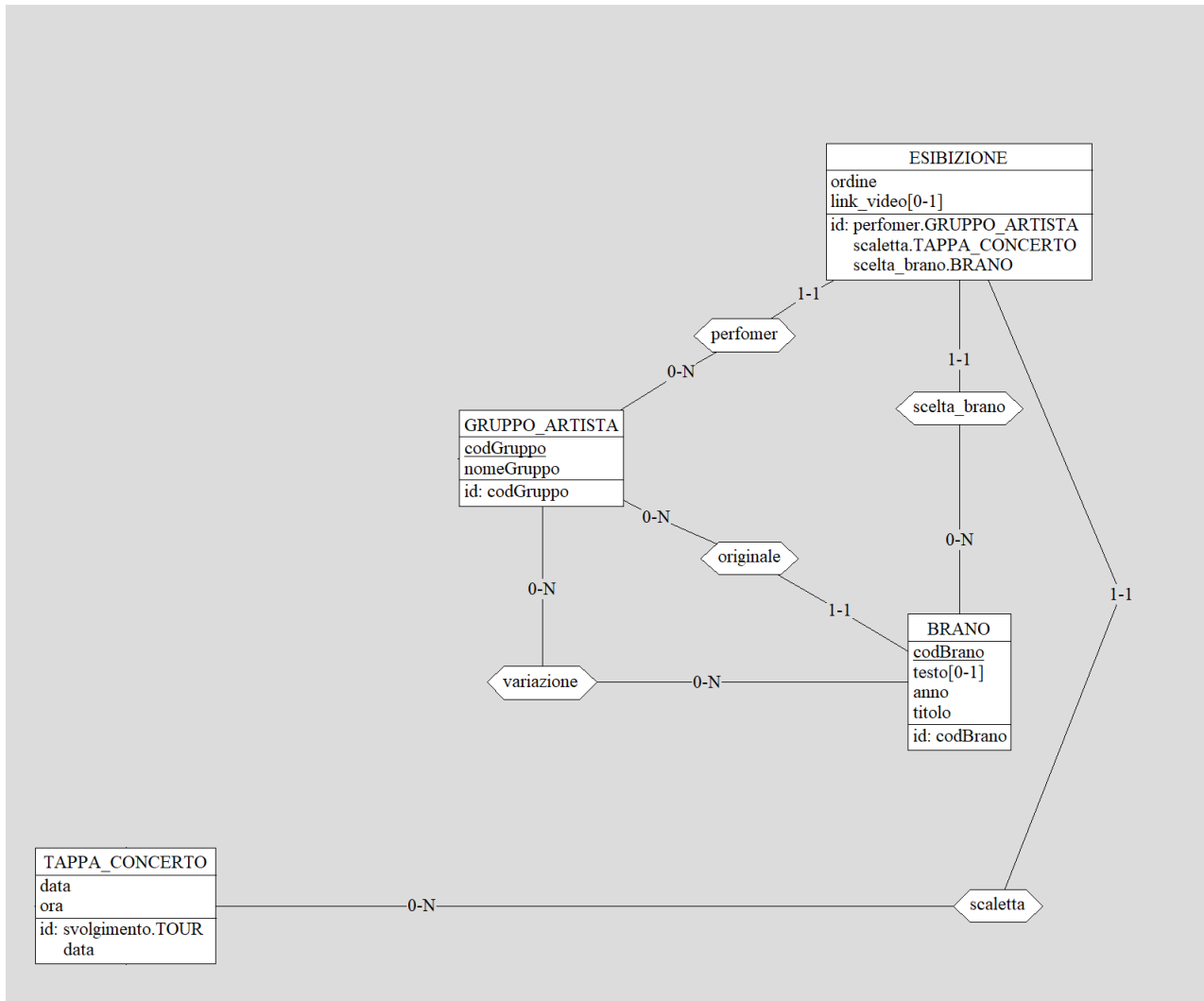
Sviluppo dell'ambito "Città-Location-Settore-Posto"



Dall'analisi emergono diverse tipologie di location ma di queste non vengono forniti dettagli pertanto non si è optato per una gerarchia ma per un semplice attributo tipo di location. Di una location si vuole conoscere la città in cui è situata e, se presenta posti numerati, la suddivisione in settori. I settori

a loro volta sono suddivisi in posti. Dal momento che possono esserci delle location con posti non numerati questo caso viene modellato come location con un unico settore.

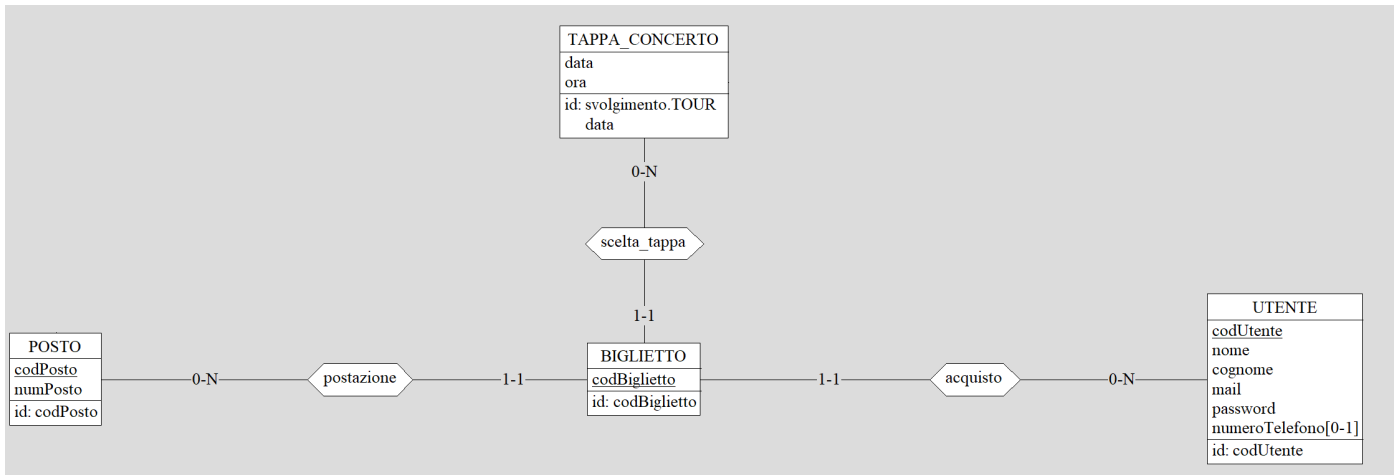
Sviluppo dell'ambito "Esibizione-Brano-Variazione"



Un concerto ha più esibizioni mentre un'esibizione fa riferimento ad un unico concerto, a un unico brano e ad un unico GRUPPO_ARTISTA. Si è scelto l'identificatore misto di ESIBIZIONE per non permettere che vi siano esibizioni ripetute all'interno dello stesso concerto. Questo inoltre rende possibile specificare l'ordine delle esibizioni tramite l'attributo ordine di cui però deve

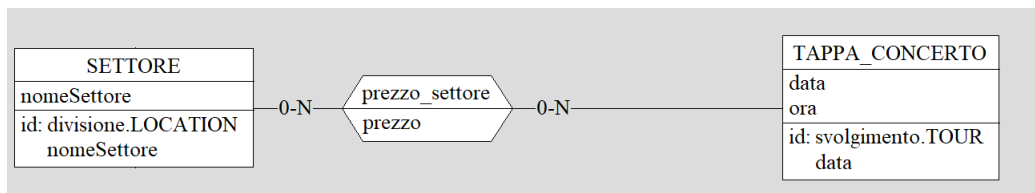
essere effettuato un corretto inserimento. Dentro ESIBIZIONE è specificato l'attributo link_video in cui è salvato, se presente, il link del video della specifica esibizione cui fa riferimento. Ovviamente un GRUPPO_ARTISTA può fare più esibizioni così come un brano può essere cantato più volte. Un brano è associato ad un solo GRUPPO_ARTISTA originale mentre a più gruppi nel caso di variazioni\remix. Si è ritenuto opportuno dare la possibilità di salvare un brano all'interno del database anche se non è stato portato in alcuna esibizione.

Sviluppo dell'ambito "Biglietto-Utente"



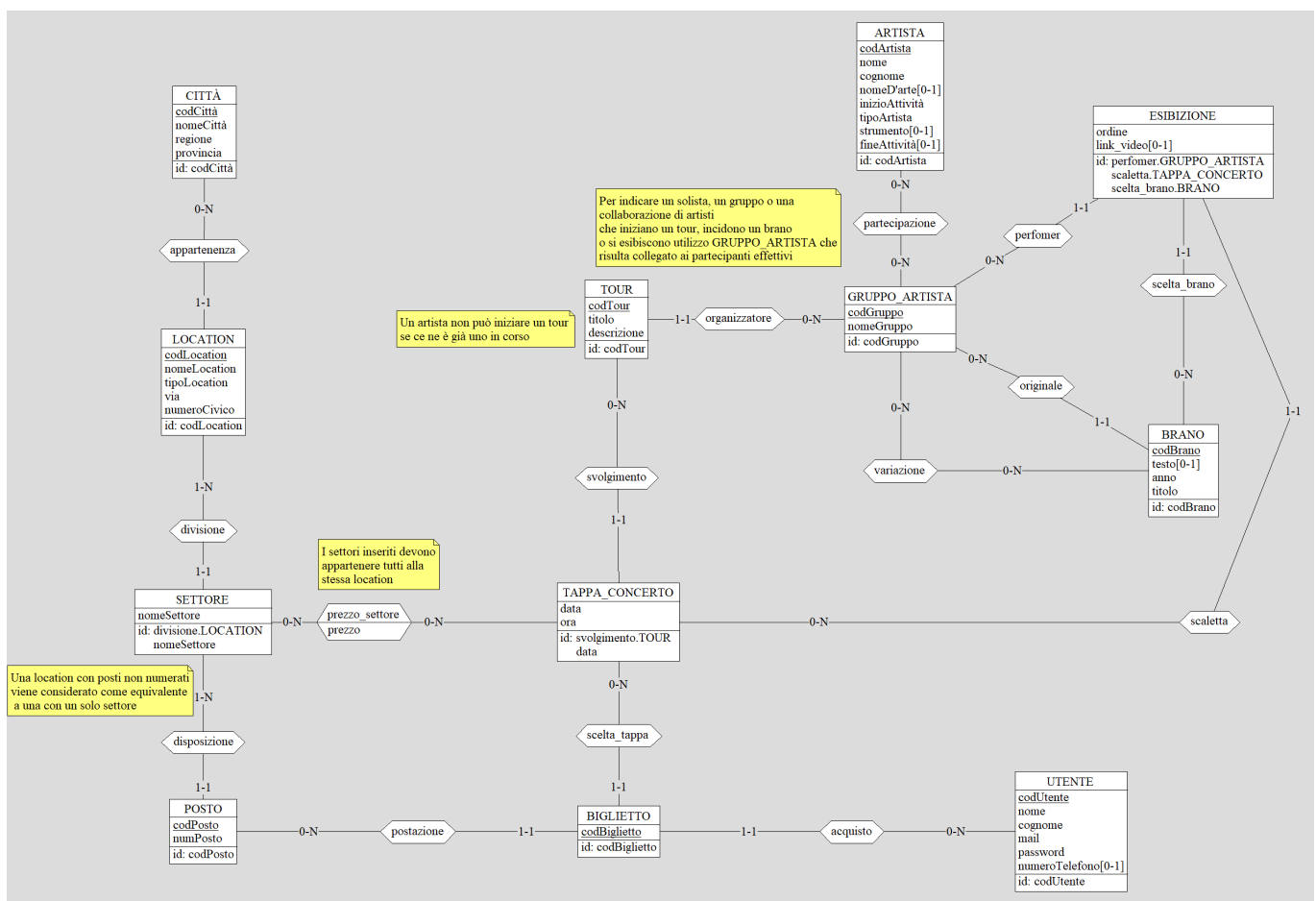
Viene modellato il concetto di Utente. Questo può acquistare più biglietti mentre un biglietto può essere acquistato da un solo utente. Un biglietto fa riferimento ad un'unica tappa e ad un unico posto opportunamente identificato.

Sviluppo dell'ambito "Prezzo Settore"



Risulta necessario collegare una TAPPA_CONCERTO con la location in cui si svolge. Dal momento che serve anche specificare il prezzo per ogni settore in base al concerto stesso è sufficiente collegare TAPPA_CONCERTO e SETTORE con la relazione prezzo_settore in modo da evitare ridondanza. Come attributo della relazione viene specificato il prezzo. Occorre assicurarsi che i settori collegati appartengano alla stessa location.

2.2 Schema concettuale finale



Capitolo 3

Progettazione logica

3.1 Stima del volume dei dati

Stima dei volumi in base a dati raccolti in un anno.

Concetto	Tipo	Volume di dati
CITTÀ	E	50
appartenenza	R	100
LOCATION	E	100
divisione	R	300
SETTORE	E	300
disposizione	R	200.000
POSTO	E	200.000
postazione	R	800.000
BIGLIETTO	E	800.000
scelta_tappa	R	800.000
TAPPA_CONCERTO	E	500
prezzo_settore	R	1500
acquisto	R	800.000
UTENTE	E	100.000
svolgimento	R	500
TOUR	E	50
organizzatore	R	50
GRUPPO_ARTISTA	E	100
partecipazione	R	50
ARTISTA	E	250

performer	R	5000
ESIBIZIONE	E	5000
scelta_branco	R	5000
BRANO	E	500
originale	R	500
variazione	R	200
scaletta	R	5000

3.2 Descrizione delle operazioni principali e stima della loro frequenza

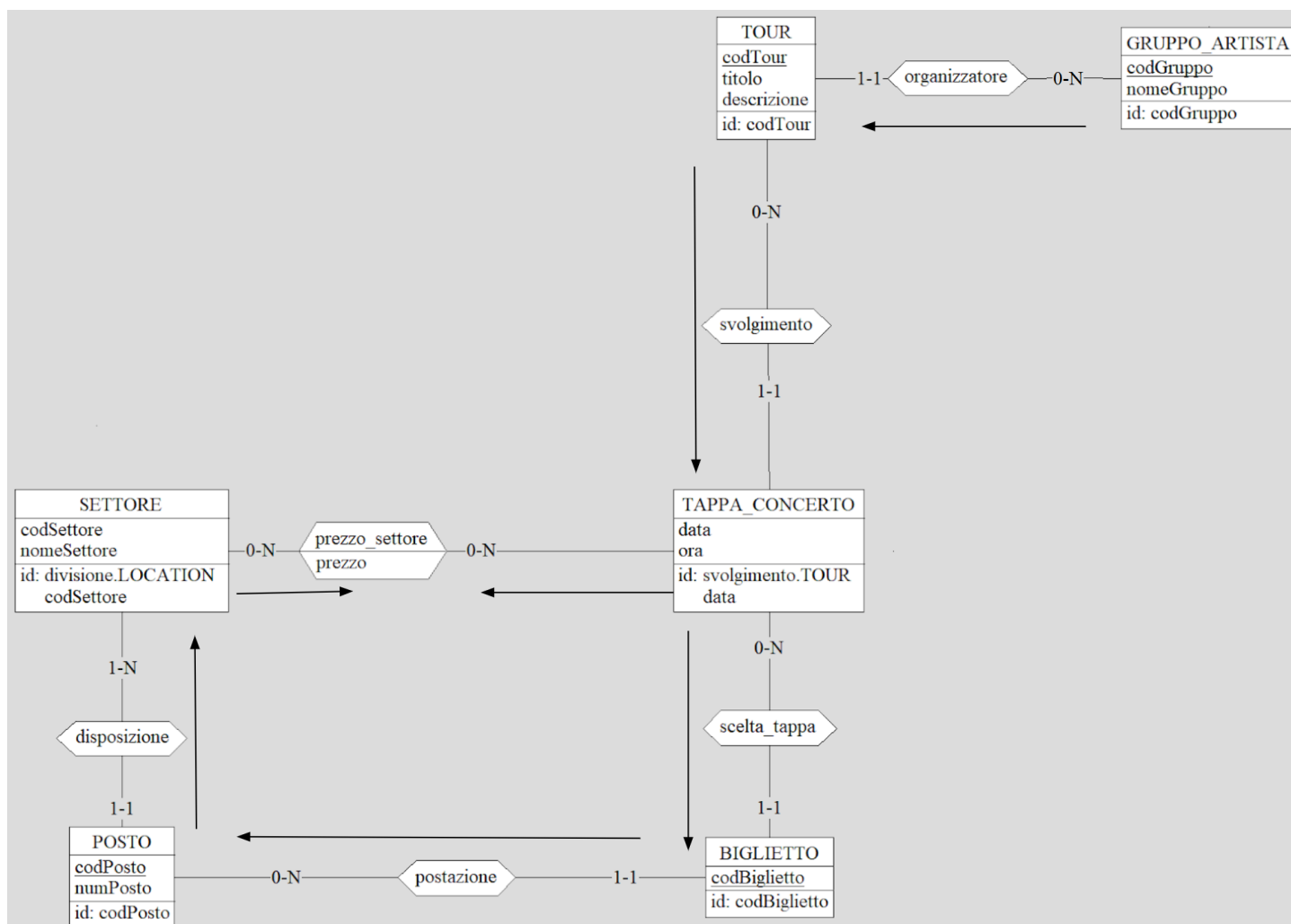
Numero	Nome Operazione	Frequenza	Tipo
1a.	Calcolo dell'incasso di tutti i tour di un artista	1/mese	B
2a.	Conteggio dei biglietti venduti per un tour	1/mese	B
3a.	Inserimento di un gruppo artista	2/anno	I
4a.	Inserimento di un prezzo settore	30/settimana	I
5a.	Inserimento di una tappa di un concerto	10/settimana	I
1u.	Visualizzazione dei prossimi concerti in una determinata città	1000/giorno	I
2u.	Visualizzazione dei prossimi concerti di un determinato artista	1000/giorno	I
3u.	Visualizzazione posti liberi	1000/giorno	I
4u.	Acquisto di un biglietto	800/giorno	I
5u.	Visualizzazione delle esibizioni di un artista di cui è presente un video	100/giorno	I
6u.	Visualizzazione della scaletta di un concerto	100/giorno	I
7u.	Visualizzazione dei biglietti acquistati dall'utente	20/giorno	I

3.3 Schemi di navigazione e tabelle degli accessi

Dopo aver determinato il volume dei dati ed aver associato a ciascuna operazione principale richiesta la propria frequenza di esecuzione, si procede determinando lo schema di navigazione di riferimento per le principali operazioni richieste e la relativa tavola degli accessi

1a. Calcolo dell'incasso di tutti i tour di un artista

Dato un artista organizzatore viene calcolato l'incasso di tutti i suoi tour.



Concetto	Costrutto	Accessi	Tipo
GRUPPO_ARTISTA	E	1	L
organizzatore	R	1	L
TOUR	E	1	L
svolgimento	R	10	L
TAPPA_CONCERTO	E	10	L
scelta_tappa	R	16000	L
BIGLIETTO	E	16000	L
postazione	R	16000	L
POSTO	E	16000	L
disposizione	R	16000	L
SETTORE	E	16000	L
prezzo_settore	R	30	L

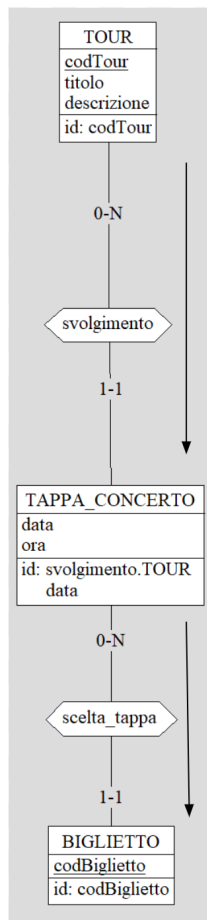
Totale: 96053 L

Frequenza operazione: 1 al mese = 0,03 al giorno

Costo totale: $96053 \times 0,03 = 2882$ al giorno

2a. Conteggio dei biglietti venduti per un tour

Dato un tour conteggio dei biglietti venduti.



Concetto	Costrutto	Accessi	Tipo
TOUR	E	1	L
svolgimento	R	10	L
TAPPA_CONCERTO	E	10	L
scelta_tappa	R	16.000	L
BIGLIETTO	E	16.000	L

Totale: 32.021L.

Frequenza operazione 1 al mese = 0,03 al giorno.

Costo totale: 961 al giorno.

3a. Inserimento di un gruppo artista

Dati n artisti creazione di un gruppo di artisti.

Concetto	Costrutto	Accessi	Tipo
GRUPPO_ARTISTA	E	1	S
partecipazione	E	n	S

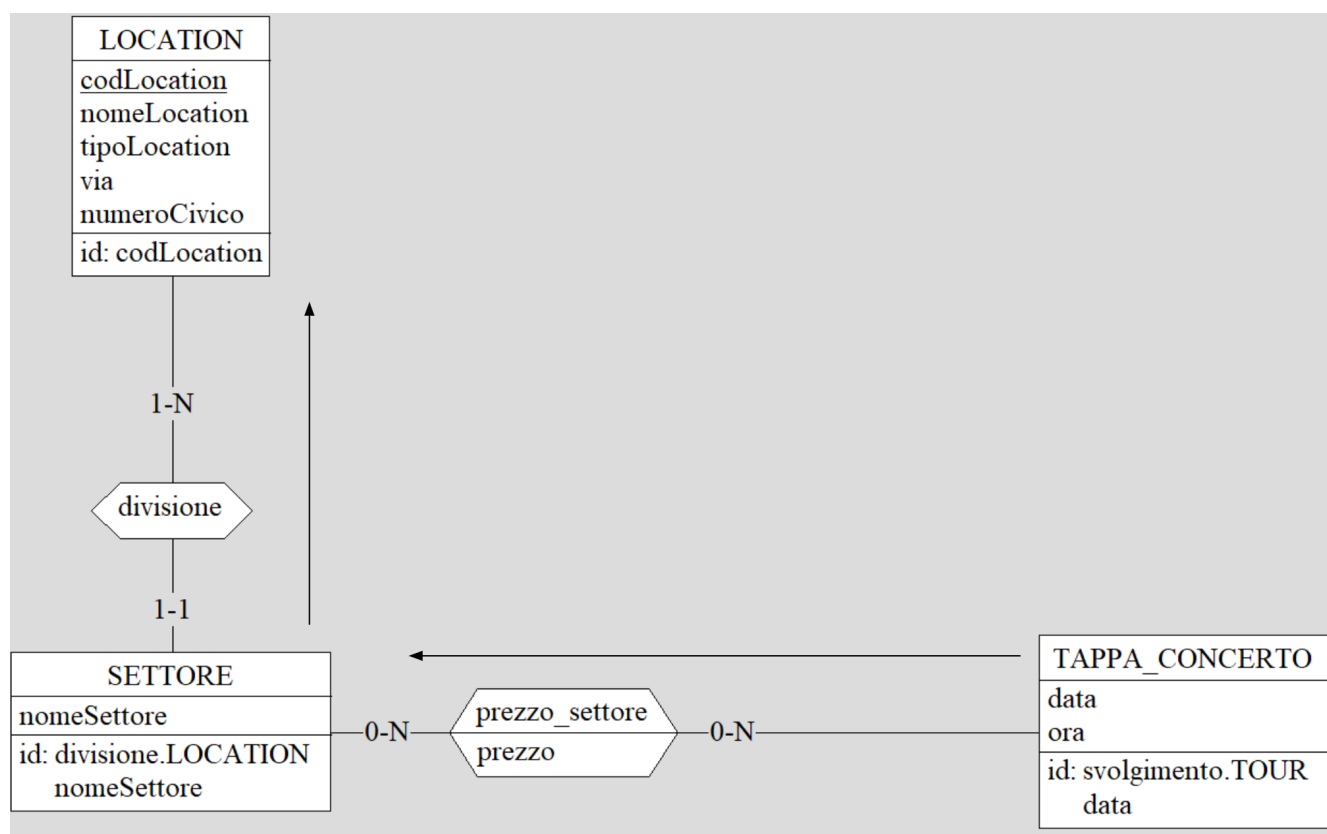
Suppongo che in media vengano aggiunti 2 artisti. Totale: $1S + 2S = 3S = 6L$

Frequenza della operazione: 2 all'anno

Costo Totale: 12 all'anno.

4a. Inserimento di un prezzo settore

Dato un concerto e un settore inserimento del prezzo relativo. Prima di inserirlo devo controllare che la location legata al settore che voglio inserire sia coerente con quella dei prezzi-settore già inseriti.



Concetto	Costrutto	Accessi	Tipo
TAPPA_CONCERTO	E	1	L
prezzo_settore	R	3	L
SETTORE	R E 3	L	
divisione	R	3	L
LOCATION	E	1	L
prezzo_settore	R	1	S

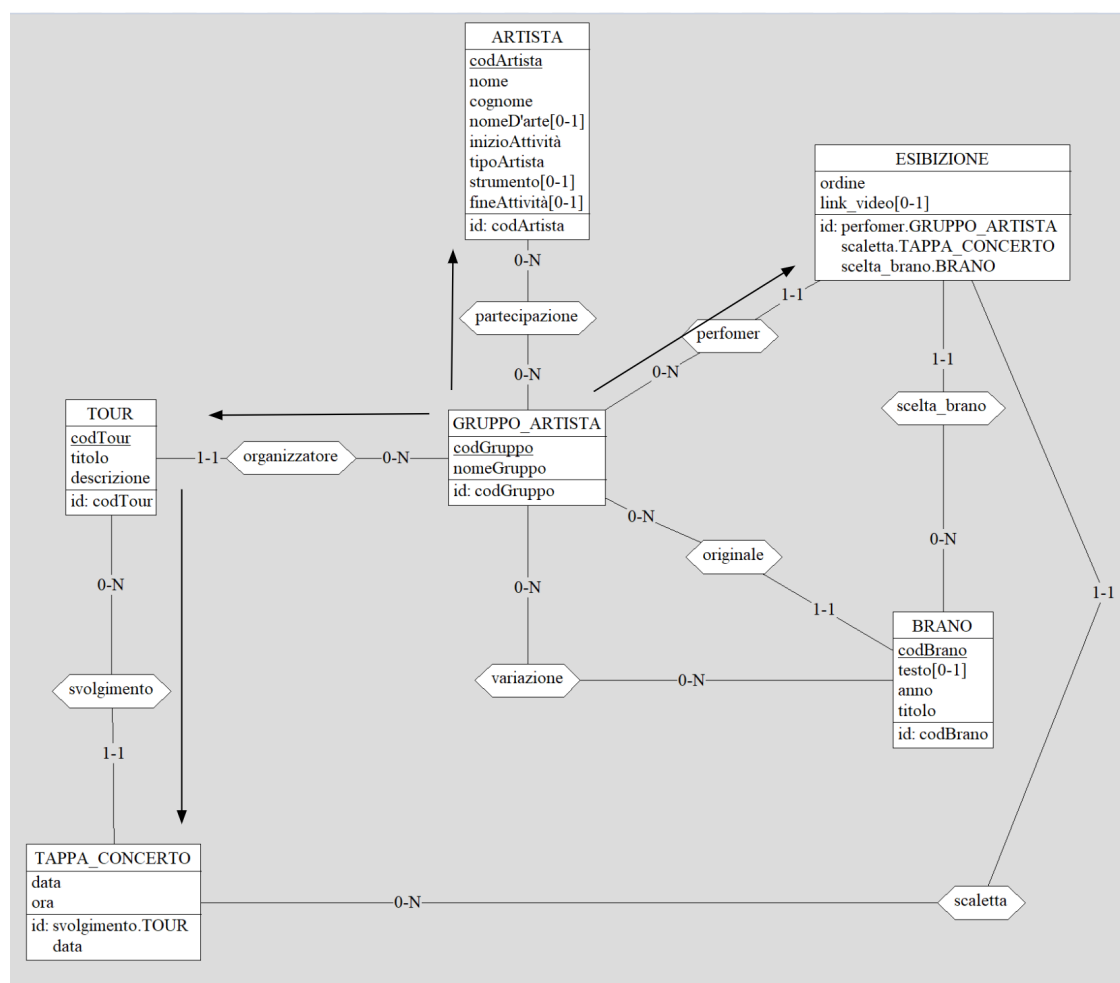
Totale = 11L + 1S = 13L

Frequenza della operazione: 30 alla settimana = 4,3 al giorno

Costo Totale: 56 al giorno.

5a. Inserimento di una tappa di un concerto

Dato un tour di un gruppo artista inserimento di una tappa. Si considera che quando viene inserita una data per un nuovo tour non possano essere più inserite date per i vecchi tour. Se si aggiunge una tappa a un tour si controlla che la data sia maggiore di quelle già presenti e si controlla che i componenti del gruppo non siano impegnati in esibizioni in quella data. Se si inserisce la prima tappa di un tour si controlla che sia maggiore delle date dei tour a cui partecipano i componenti del gruppo e si verificano le esibizioni come nel caso precedente.



Concetto	Costrutto	Accessi	Tipo
ARTISTA	E	2,5	L
GRUPPO_ARTISTA	E	1,4	L
performer	R	70	L
ESIBIZIONE	E	70	L
TOUR	E	0,7	L
svolgimento	R	10	L
TAPPA_CONCERTO	E	11	L
TAPPA_CONCERTO	E	1	S

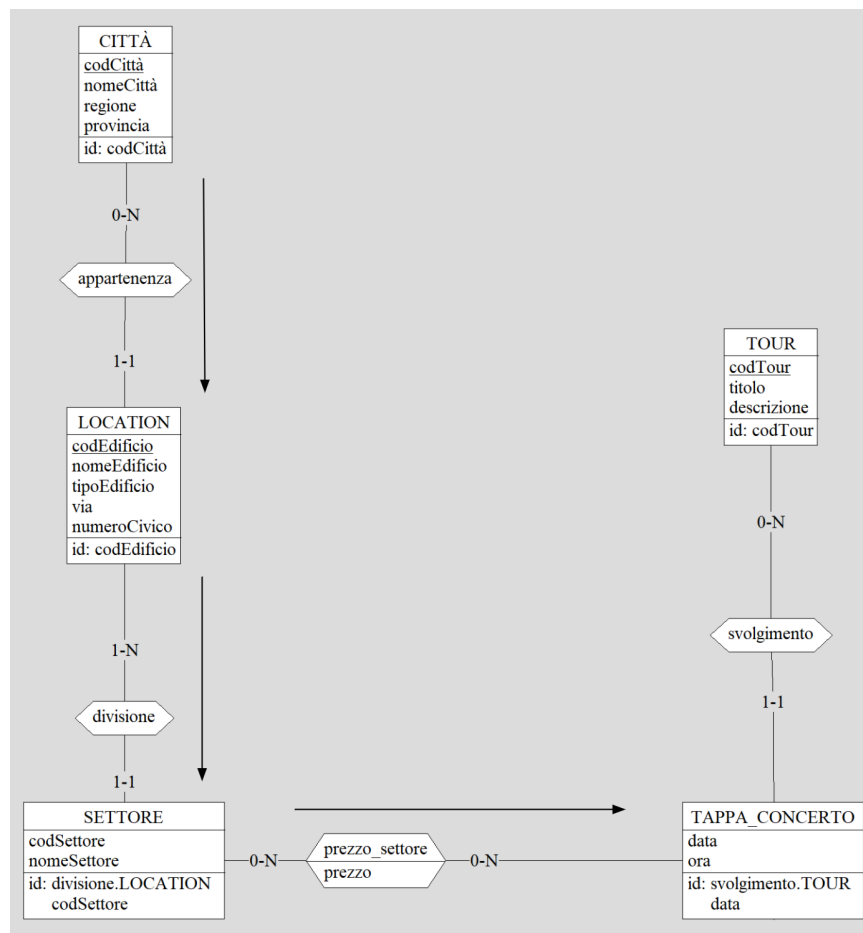
Totale: $166L + 1S = 168$.

Frequenza della operazione: 10 alla settimana = 1,43 al giorno

Costo totale: 240 al giorno.

1u. Visualizzazione dei concerti in una determinata città

Data una città visualizzazione dei concerti che vi si terranno.



Concetto	Costrutto	Accessi	Tipo
CITTÀ	E	1	L
appartenenza	R	2	L
LOCATION	E	2	L
divisione	R	6	L
SETTORE	E	6	L
prezzo_settore	R	30	L
TAPPA_CONCERTO	E	30	L

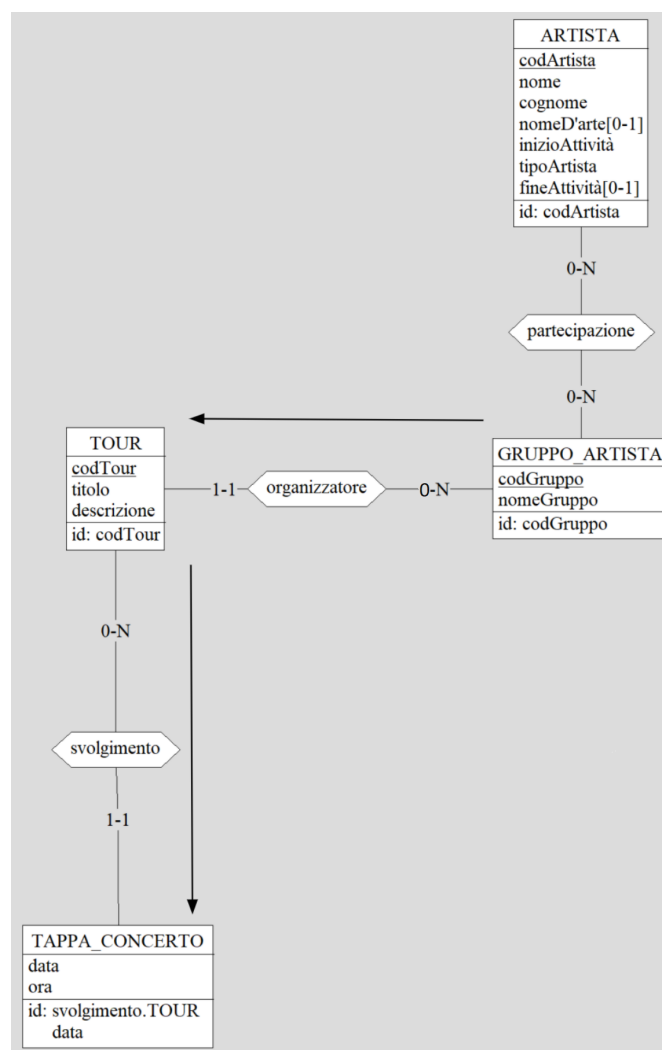
Totale: 77L

Frequenza operazione: 1000 al giorno.

Costo totale: 77.000 al giorno.

2u. Visualizzazione dei concerti di un determinato artista

Dato un artista organizzatore visualizzazione dei concerti che si terranno nell'anno corrente.



Concetto	Costrutto	Accessi	Tipo
GRUPPO_ARTISTA	E	1	L
organizzatore	R	1	L
TOUR	E	1	L
TAPPA_CONCERTO	E	10	L

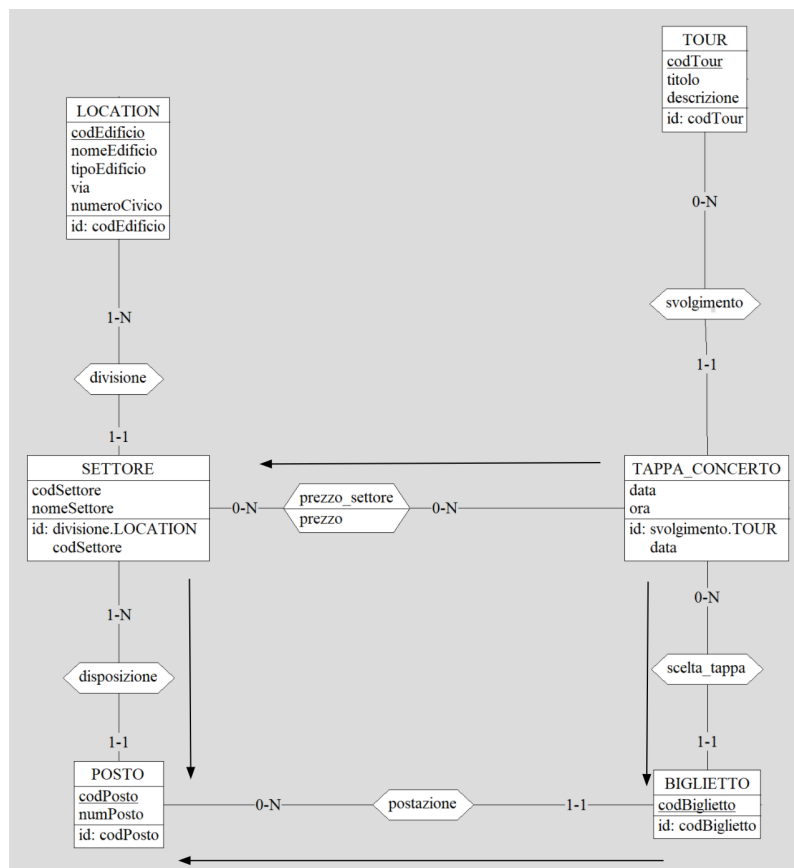
Totale: 13L

Frequenza della operazione: 1000 al giorno.

Costo totale: 1000 x 13 = 13.000 al giorno.

3u. Visualizzazione posti liberi

Dato un concerto visualizzazione del numero di posti liberi per settore con relativo prezzo.



Concetto	Costrutto	Accessi	Tipo
TAPPA_CONCERTO	E	1	L
scelta_tappa	R	1600	L
BIGLIETTO	E	1600	L
postazione	E	1600	L
prezzo_settore	E	3	L
POSTO	E	1600	L
SETTORE	E	3	L
disposizione	R	1600	L

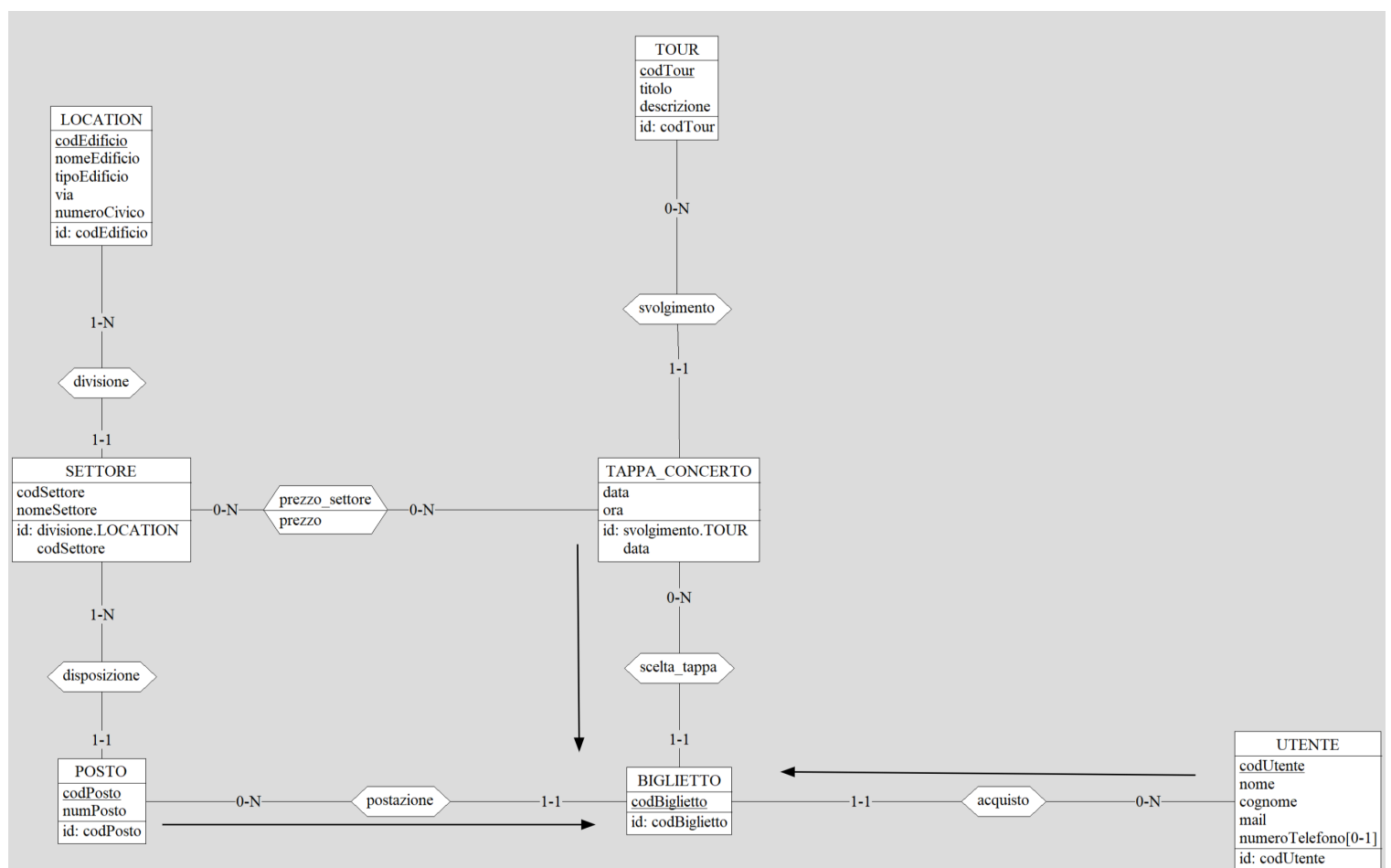
Totale: 6407.

Frequenza operazione: 1000 al giorno.

Costo totale: 6.407.000 al giorno.

4u. Acquisto di un biglietto

Dato un concerto, un posto libero (il controllo è considerato come operazione 3u) e un utente, acquisto di un biglietto.



Concetto	Costrutto	Accessi	Tipo
BIGLIETTO	E	1	S
scelta_tappa	R	1	S
postazione	R	1	S
acquisto	R	1	S

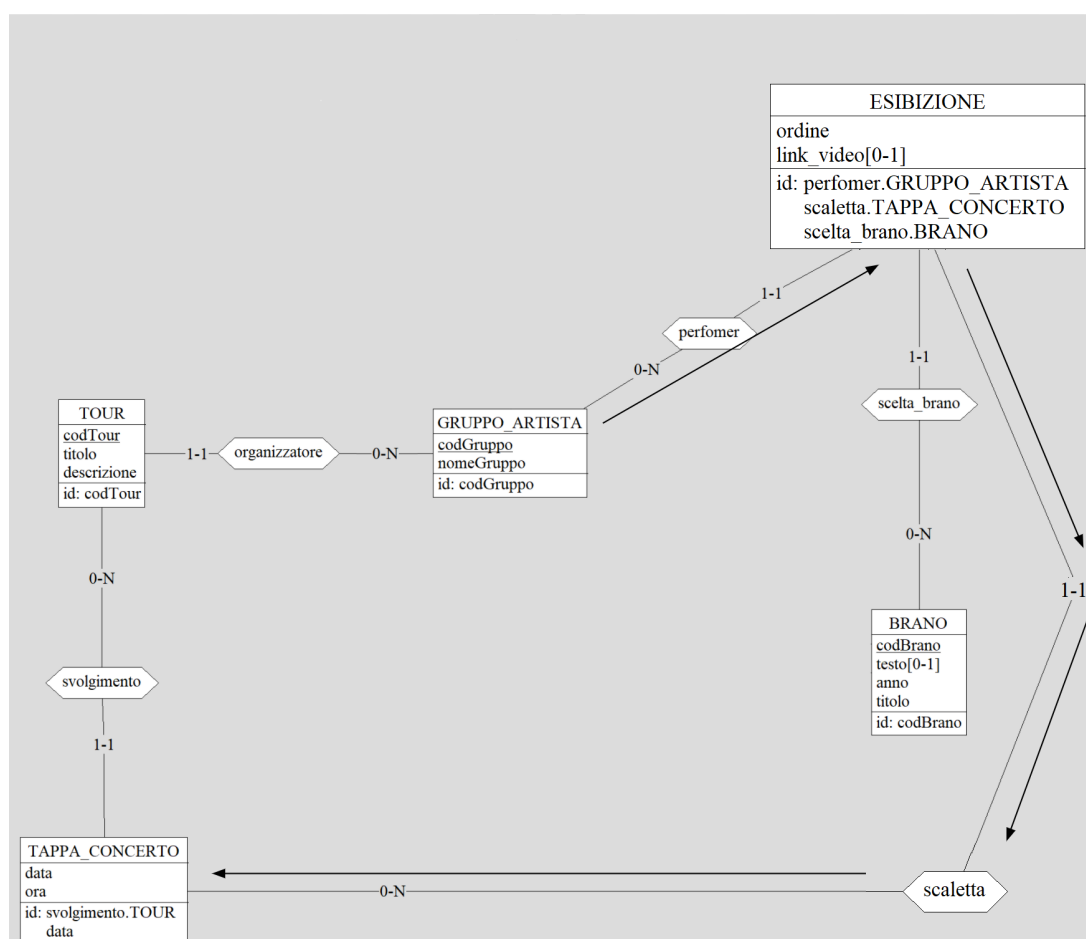
Totale: $4S = 2 \times 4L = 8L$.

Frequenza operazione: 800 al giorno.

Costo totale: 6400 al giorno.

5u. Visualizzazione delle esibizioni di un artista di cui c'è un video

Dato un gruppo di artisti, visualizzazione delle informazioni delle esibizioni a cui è associato un video indicando la tappa a cui si riferisce.



Concetto	Costrutto	Accessi	Tipo
GRUPPO _ARTISTA	E	1	L
performer	R	50	L
ESIBIZIONE	E	50	L
scaletta	R	50	L
TAPPA _CONCERTO	E	50	L

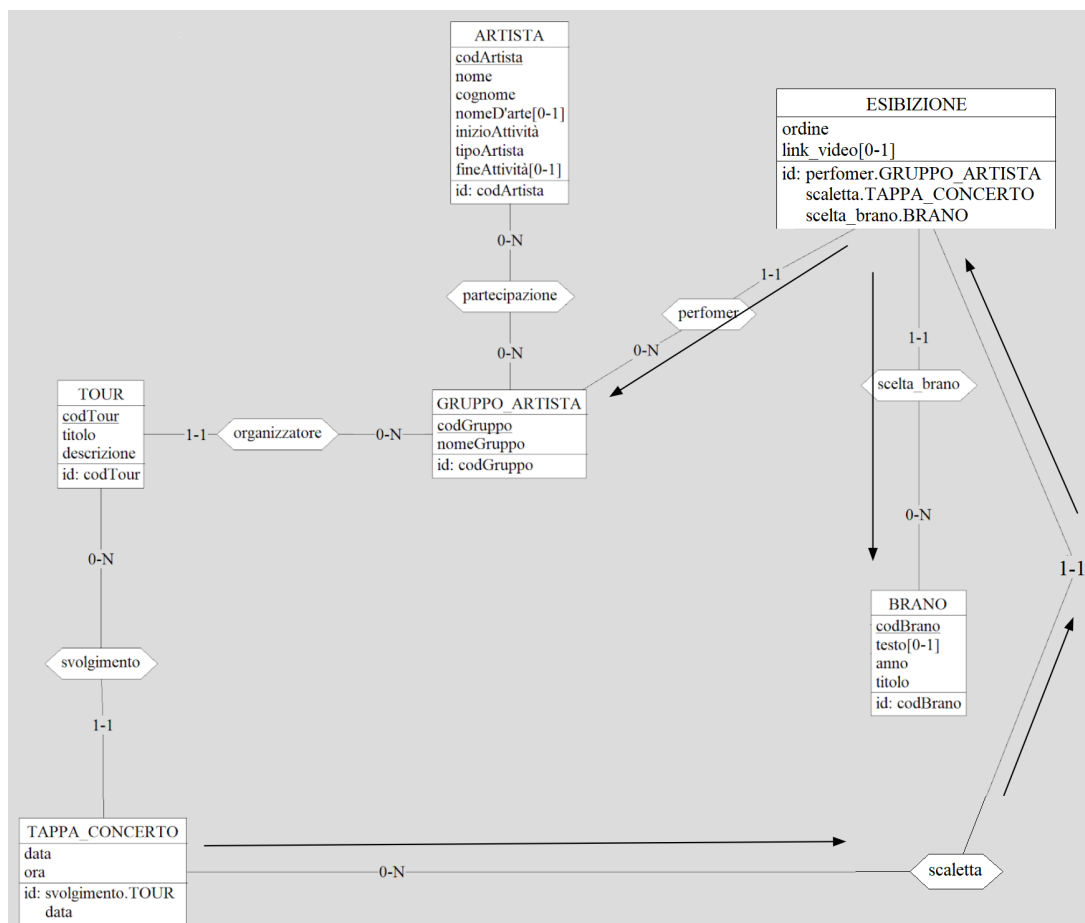
Totale: 201L.

Frequenza operazione: 100 al giorno.

Costo totale: 20.100 al giorno.

6u. Visualizzazione scaletta

Dato un concerto visualizzazione della scaletta, in ordine saranno indicati gli artisti e i brani che vorranno presentare.



Concetto	Costrutto	Accessi	Tipo
TAPPA_CONCERTO	E	1	L
scaletta	R	10	L
ESIBIZIONE	E	10	L
scelta_branco	R	10	L
BRANO	E	10	L
performer	R	10	L
GRUPPO_ARTISTA	R	10	L

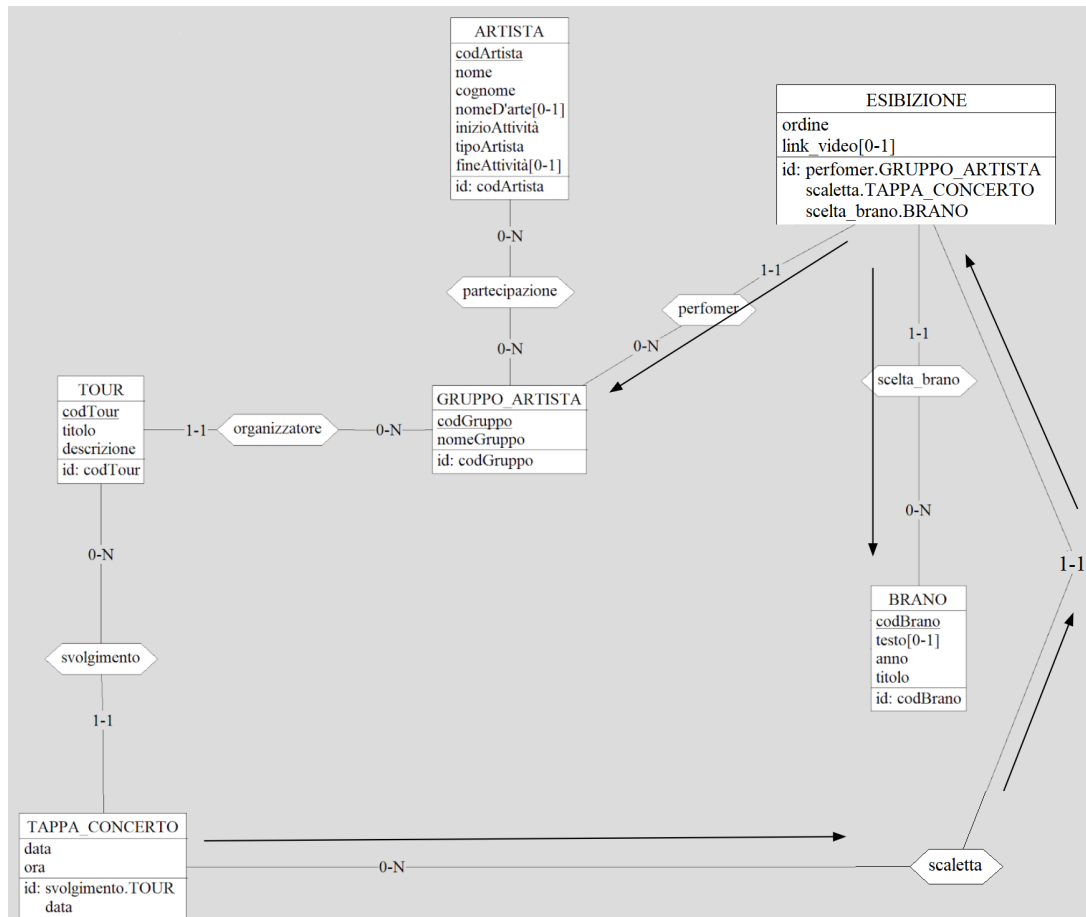
Totale: 61L.

Frequenza operazione: 100 al giorno.

Costo totale: 6.100 al giorno.

7u. Visualizzazione dei biglietti acquistati dall'utente

Dato un utente sono visualizzati i biglietti acquistati



Concetto	Costrutto	Accessi	Tipo
UTENTE	E	1	L
acquisto	R	8	L
BIGLIETTO	E	10	L

Totale: 19L.

Frequenza operazione: 20 al giorno.

Costo totale: 20x19=380 al giorno.

3.4 Analisi delle ridondanze

Le operazioni più costose che richiederebbero un'ottimizzazione sono 1u, 2u, 3u, 5u. Le operazioni 1u, 2u, 5u non sono ottimizzabili tramite ridondanza in quanto non prevedono raggruppamenti ma si occupano di visualizzare grandi quantità di dati. Si potrebbero migliorare mediante l'inserimento di un indice. Di seguito studio la variazione del costo dell'operazione 3u introducendo l'attributo ridondante numLiberi all'interno della relazione prezzo_settore. L'attributo inserito indica il numero di posti ancora liberi per ogni coppia settore-tappa_concerto.

L'aggiunta di questa ridondanza influenza anche il costo dell'operazione 9.

Studio del costo dell'operazione 3u.

Concetto	Costrutto	Accessi	Tipo
TAPPA_CONCERTO	E	1	L
prezzo_settore	R	3	L
SETTORE	E	3	L

Totale: 7L.

Frequenza operazione: 1.000 al giorno.

Costo totale: 7.000 al giorno.

Studio del costo dell'operazione 4u.

Concetto	Costrutto	Accessi	Tipo
BIGLIETTO	E	1	S
postazione	R	1	S
acquisto	R	1	S
scelta_tappa	R	1	S
TAPPA_CONCERTO	E	1	L
prezzo_settore	R	1	L
prezzo_settore	R	1	S

Totale: $2L + 5S = 2 + 2 \times 5 = 12$.

Frequenza operazione: 800 al giorno.

Costo totale: 9.600 al giorno.

Con ridondanza abbiamo $9.600 + 7.000 = 16.600$ invece di $6.400 + 6.407.000 = 6.413.400$.

Risulta molto più conveniente con l'introduzione della ridondanza.

3.5 Raffinamento dello schema

3.5.1 Eliminazione delle gerarchie

All'interno dello schema non sono presenti gerarchie.

3.5.2 Scelta delle chiavi primarie

Per quanto riguarda l'entità BIGLIETTO si è ritenuto opportuno utilizzare un codice come identificatore per rendere più semplici e veloci le operazioni che la coinvolgono.

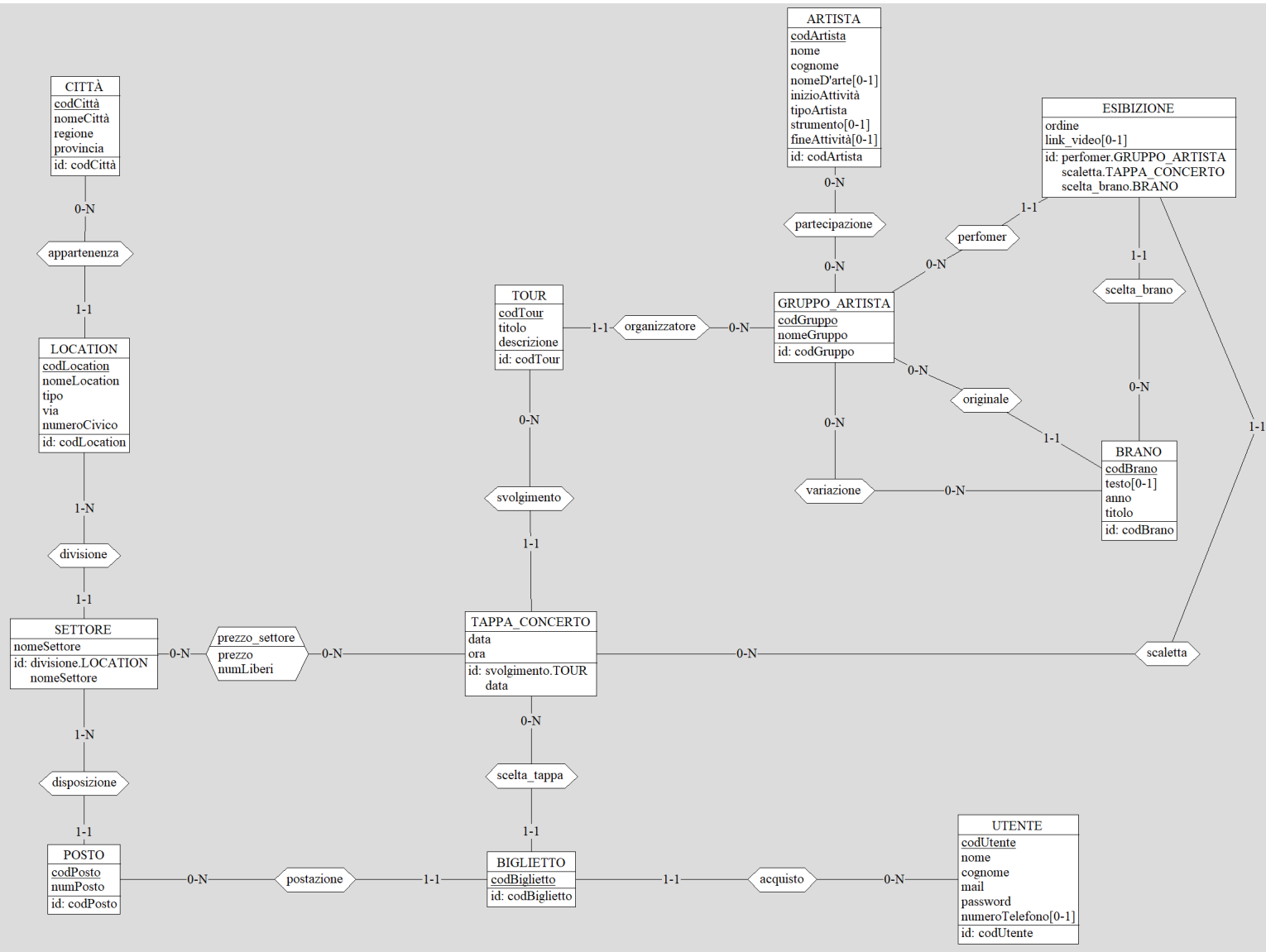
Dal momento che il nome di un settore risulta univoco solo in base alla location in cui si trova si è deciso di identificare l'entità SETTORE tramite un identificatore composto dal nome del settore e dalla LOCATION.

Per l'entità POSTO non si è adottata la stessa soluzione per rendere le operazioni più veloci e evitare di avere un identificatore troppo grande.

TAPPA_CONCERTO è identificata mediante la data e il TOUR di appartenenza in quanto il solo uso di data risulterebbe troppo vincolante.

Per ESIBIZIONE si utilizza un identificatore composto che risulta più adatto ad esprimere il vincolo per cui all'interno di uno stesso concerto non possa esserci più volte un brano cantato da un determinato artista.

3.6 Schema concettuale con aggiunta di ridondanza



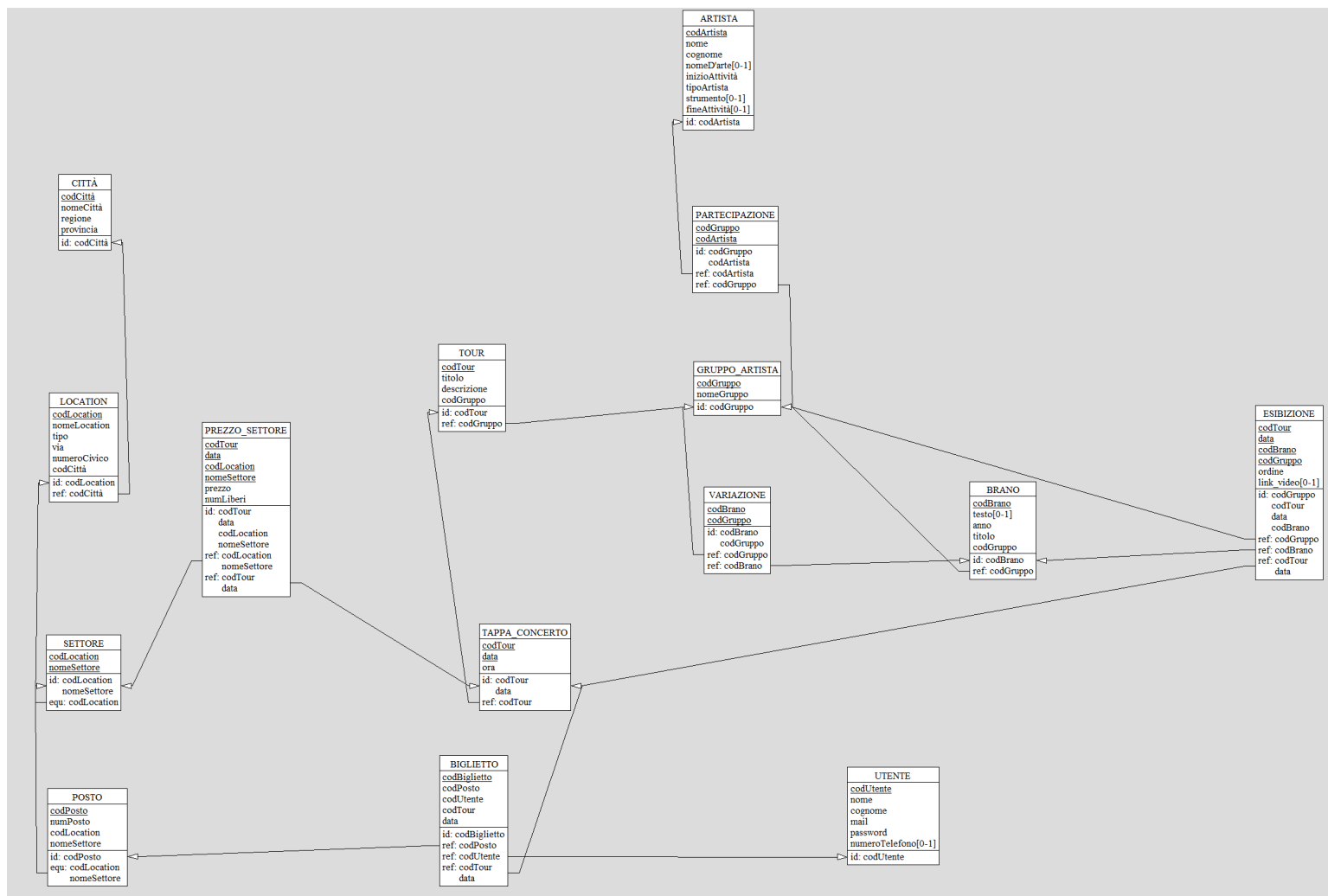
3.7 Traduzione di entità e associazioni in relazioni

- **CITTÀ**(codCittà, nomeCittà, regione, provincia)

- **LOCATION**(codLocation, nomeLocation, tipo, via, numeroCivico, codCittà)
FK: codCittà REFERENCES **CITTÀ**
- **SETTORE**(nomeSettore, codLocation)
FK: codLocation REFERENCES **LOCATION**
- **POSTO**(codPosto, numPosto, nomeSettore, codLocation)
FK: nomeSettore REFERENCES **SETTORE**
FK: codLocation REFERENCES **LOCATION**
- **ARTISTA**(codArtista, nome, cognome, nomeD'arte *, inizioAttività, tipoArtista, strumento *, fineAttività *)
- **PARTECIPAZIONE**(codGruppo, codArtista)
FK: codGruppo REFERENCES **GRUPPO _ ARTISTA**
FK: codArtista REFERENCES **ARTISTA**
- **GRUPPO _ ARTISTA**(codGruppo, nomeGruppo)
- **TOUR**(codTour, titolo, descrizione, codGruppo)
FK: codGruppo REFERENCES **GRUPPO _ ARTISTA**
- **TAPPA _ CONCERTO**(data, codTour,ora)
FK: codTour REFERENCES **TOUR**
- **UTENTE**(codUtente, nome, cognome, mail, password, numeroTelefono *)
- **BIGLIETTO**(codBiglietto, codPosto, codUtente, data, codTour)
FK: codPosto REFERENCES **POSTO**
FK: codUtente REFERENCES **UTENTE**
FK: data REFERENCES **TAPPA _ CONCERTO**
FK: codTour REFERENCES **TOUR**

- **BRANO**(codBrano, testo*, anno, titolo, codGruppo)
FK: codGruppo REFERENCES **GRUPPO _ ARTISTA**
- **VARIAZIONE**(codBrano, codGruppo)
FK: codBrano REFERENCES **BRANO**
FK: codGruppo REFERENCES **GRUPPO _ ARTISTA**
- **ESIBIZIONE**(codGruppo, codBrano, data, codTour, ordine, link _ video*)
FK: codGruppo REFERENCES **GRUPPO _ ARTISTA**
FK: codBrano REFERENCES **BRANO**
FK: data REFERENCES **TAPPA _ CONCERTO**
FK: codTour REFERENCES **TOUR**
- **PREZZO _ SETTORE**(nomeSettore, codLocation, data, codTour, prez-
zo, numLiberi)
FK: nomeSettore REFERENCES **SETTORE**
FK: codLocation REFERENCES **LOCATION**
FK: data REFERENCES **TAPPA _ CONCERTO**
FK: codTour REFERENCES **TOUR**

3.8 Schema logico finale



3.9 Traduzione delle operazioni in query SQL

1a. Calcolo dell'incasso di tutti i tour di un artista

```

SELECT T.titolo, SUM (INCASSI.incassoSettore) AS incassoTour
FROM TOUR T JOIN (SELECT PS.codTour, PS.data, PS.codLocation,
PS.nomeSettore, (PS.prezzo*(PT.totalePosti-PS.numLiberi)) AS incassoSettore
FROM PREZZO_SETTORE PS JOIN (SELECT SETTORE.nomeSettore,

```

```

SETTORE.codLocation, COUNT(*) AS totalePosti
FROM SETTORE JOIN POSTO ON (SETTORE.nomeSettore=POSTO.nomeSettore
and SETTORE.codLocation=POSTO.codLocation)
GROUP BY SETTORE.nomeSettore, SETTORE.codLocation ) PT ON
(PS.nomeSettore=PT.nomeSettore and PS.codLocation=PT.codLocation))
INCASSI ON (T.codTour = INCASSI.codTour)
WHERE T.codGruppo=v1
GROUP BY T.codTour, T.titolo
ORDER BY T.titolo DESC

```

2a. Conteggio dei biglietti venduti per un tour

```

SELECT COUNT(*) AS numBiglietti
FROM TOUR T JOIN BIGLIETTO B ON (T.codTour = B.codTour)
WHERE T.codTour=v1

```

3a. Inserimento di un gruppo artista

Devo inserire il nuovo gruppo artista.

```

INSERT INTO GRUPPO_ARTISTA(codGruppo,nomeGruppo)
VALUES(v1, v2)

```

Devo inserire i singoli artisti in PARTECIPAZIONE

```

INSERT INTO PARTECIPAZIONE(codGruppo,codArtista)
VALUES(v1,v3)

```

4a. Inserimento di un prezzo settore

Data una tappa esistente controllo che la location del prezzo settore che voglio inserire sia coerente. Se la query non restituisce record posso procedere con l'inserimento

```

SELECT *
FROM PREZZO_SETTORE PS
WHERE codTour = v1 and data = v2 and codLocation != v3

```

Devo calcolare la variabile numLiberi con la seguente query.

```

SELECT COUNT(*) as numLiberi
FROM POSTO P
WHERE P.codLocation = v3 and P.nomeSettore = v4

```

GROUP BY P.codLocation, P.nomeSettore

Per l'inserimento:

```
INSERT INTO PREZZO_SETTORE(codTour, data, codLocation, nomeSet-
tore, prezzo, numLiberi)
VALUES(v1,v2,v3,v4,v5,v6)
```

4a. Inserimento di una tappa concerto

Dato un tour devo inserire un concerto e questa operazione si suddivide in diverse fasi di controllo.

1. Guardo gli artisti che fanno parte del gruppo:

```
SELECT P.codArtista
FROM PARTECIPAZIONE P
WHERE P.codGruppo = v1
```

2. Per ognuno di questi artisti devo prendere in considerazione i gruppi di cui fa parte:

```
SELECT DISTINCT P.codGruppo
FROM PARTECIPAZIONE P JOIN (SELECT P.codArtista
FROM PARTECIPAZIONE P
WHERE P.codGruppo = v1) AS AR ON (AR.codArtista = P.codArtista)
```

3. Controllo se il tour che considero ha già delle tappe inserite:

```
SELECT *
FROM TAPPA_CONCERTO TC
WHERE TC.codTour = v2
```

4a. Se non ci sono date controllo se gli artisti sono impegnati in esibizioni in quella data

Nella sottoquery viene usata la query del punto 2. Se risulta vuota posso inserire

```
SELECT *
FROM ESIBIZIONE E JOIN (SELECT DISTINCT P.codGruppo FROM
PARTECIPAZIONE P JOIN (SELECT P.codArtista FROM PARTECI-
PAZIONE P WHERE P.codGruppo = v1) AS AR ON (AR.codArtista =
P.codArtista)) AS GRUPPI ON (GRUPPI.codGruppo = E.codGruppo)
```

4b. Dato che ha delle tappe devo guardare se è il tour tra quelli che coinvol-

gono gli artisti del gruppo del tour che voglio inserire) con la data più recente, cioè il tour di cui posso aggiungere date. Notare che viene usata la query del punto 2 come sottoquery.

```
SELECT TOP 1 TC.codTour, TC.data
FROM (TAPPA_CONCERTO TC JOIN TOUR T ON (TC.codTour =
T.codTour)) JOIN ((SELECT DISTINCT P.codGruppo FROM PARTECI-
PAZIONE P JOIN (SELECT P.codArtista FROM PARTECIPAZIONE P
WHERE P.codGruppo = v1) AS AR ON (AR.codArtista = P.codArtista)))
AS GRUPPI ON (TC.codTour = T.codTour and GRUPPI.codGruppo =
T.codGruppo) ORDER BY TC.data DESC
```

Se la query 4b dà come risultato il codice del tour che voglio inserire e una data minore di quella che voglio inserire allora devo controllare la 4a che se vuota mi permette di inserire:

5. Inserisco:

```
INSERT INTO TAPPA_CONCERTO
VALUES(v2,v3,v4);
```

1u. Visualizzazione dei prossimi concerti in una determinata città

```
SELECT DISTINCT T.titolo, GA.nomeGruppo, TC.data
FROM (((CITTA C JOIN LOCATION L ON (C.codCitta = L.codCitta)
JOIN PREZZO_SETTORE PS ON (PS.codLocation = L.codLocation))
JOIN TAPPA_CONCERTO TC ON (TC.codTour = PS.codTour and TC.data
= PS.data))) JOIN TOUR T ON (T.codTour = TC.codTour)) JOIN GRUP-
PO_ARTISTA GA ON (GA.codGruppo = T.codGruppo)
WHERE C.nomeCitta=v1 and PS.data >= SYSDATETIME()
```

2u. Visualizzazione dei prossimi concerti di un artista

```
SELECT DISTINCT T.titolo, GA.nomeGruppo, TC.data
FROM (((CITTA C JOIN LOCATION L ON (C.codCitta = L.codCitta)
JOIN PREZZO_SETTORE PS ON (PS.codLocation = L.codLocation))
JOIN TAPPA_CONCERTO TC ON (TC.codTour = PS.codTour and TC.data
= PS.data))) JOIN TOUR T ON (T.codTour = TC.codTour)) JOIN GRUP-
PO_ARTISTA GA ON (GA.codGruppo = T.codGruppo)
WHERE Ga.nomeGruppo=v1 and PS.data >= SYSDATETIME()
```

3u. Visualizzazione posti liberi per settore

```
SELECT DISTINCT PS.nomeSettore, PS.prezzo, PS.numLiberi  
FROM PREZZO_SETTORE PS  
WHERE PS.data=v1 and PS.codTour=v2
```

4u. Acquisto di un biglietto

Inserimento del nuovo biglietto.

```
INSERT INTO BIGLIETTO(codBiglietto, codPosto, codUtente,codTour,data)  
VALUES(v1,v2,v3,v4,v5)
```

Aggiornamento di prezzo settore.

```
UPDATE PREZZO_SETTORE  
SET numLiberi=numLiberi-1  
WHERE codTour=v4 and data=v5 and codLocation=v6 and nomeSettore=v7
```

5u. Visualizzazione delle esibizioni di un artista in cui è presente un video

```
SELECT DISTINCT GA.nomeGruppo, B.titolo, E.data  
FROM (GRUPPO_ARTISTA GA JOIN ESIBIZIONE E ON (GA.codGruppo  
= E.codGruppo)) JOIN BRANO B ON (E.codBrano = B.codBrano and  
E.codGruppo=B.codGruppo)  
WHERE GA.codGruppo=v1 and E.link_video IS NOT NULL
```

6u. Visualizzazione della scaletta di un concerto

```
SELECT DISTINCT GA.nomeGruppo, B.titolo, E.ordine  
FROM (GRUPPO_ARTISTA GA JOIN ESIBIZIONE E ON (GA.codGruppo  
= E.codGruppo)) JOIN BRANO B ON (E.codBrano = B.codBrano and  
E.codGruppo=B.codGruppo)  
WHERE E.codTour=v1 and E.data=v2  
ORDER BY E.ordine
```

7u. Visualizzazione dei biglietti acquistati dall'utente

```
SELECT DISTINCT GA.nomeGruppo, T.titolo, B.data, C.nomeCitta, L.nomeLocation,  
P.nomeSettore, P.numPosto  
FROM ((GRUPPO_ARTISTA GA JOIN TOUR T ON (GA.codGruppo=T.codGruppo))  
JOIN BIGLIETTO B ON (B.codTour = T.codTour)) JOIN POSTO P ON  
(B.codPosto=P.codPosto) JOIN LOCATION L ON (P.codLocation=L.codLocation)  
JOIN CITTA C ON (L.codCitta =C.codCitta) WHERE B.codUtente = v1
```


Capitolo 4

Progettazione dell'applicazione

4.1 Descrizione dell'architettura dell'applicazione

L'applicazione per interfacciarsi al database è stata realizzata in C# e fa uso di LINQ. Il database risiede in locale e il DBMS usato è SQL server.

Attraverso il piccolo menù principale si può effettuare l'accesso come amministratore o come utente per il quale viene richiesta l'autenticazione

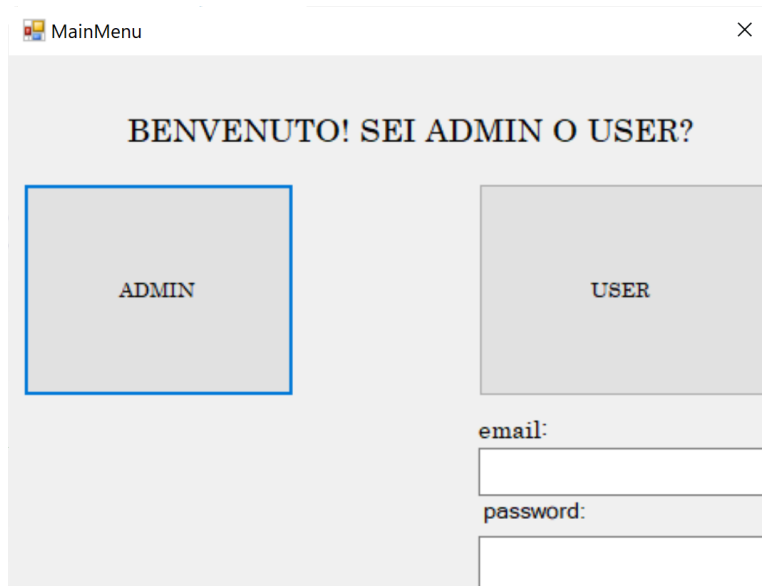


Figura 4.1: Menù principale

Dall'interfaccia dell'amministratore è possibile effettuare le operazioni del tipo "a". Tramite menù a tendina in alto a destra è possibile selezionare l'operazione e una volta fatto questo sarà possibile selezionare i dati tramite le griglie e inserire dati tramite le caselle per l'input opportune. La griglia a destra viene usata per visualizzare l'output.

FormAdmin

Concerti

	codTour	data	titoloTour	nomeGruppo
▶	2	21/01/2021	This is Elodie	Elodie
	2	12/05/2021	This is Elodie	Elodie
	4	28/02/2021	Divisions	Starset
	4	15/06/2021	Divisions	Starset

Inserimento di prezzi relativi a tappa e settore

Prezzo

Continua

Settori

	codLocation	nomeSettore	nomeLocation
▶	4	1	Unipol Arena
	4	2	Unipol Arena
	3	platea	Alcatraz
	1	A	Mediolanum Foru
	1	B	Mediolanum Foru
	1	C	Mediolanum Foru
	2	palco	Teatro Centrale
	2	platea	Teatro Centrale

Figura 4.2: Interfaccia Amministratore

L'interfaccia utente è stata sviluppata seguendo un approccio simile a quello adottato per l'interfaccia dell'amministratore, preferendo l'uso di menù a tendina e con una griglia per la visualizzazione dei risultati.

FormUser

Visualizzazione dei posti liberi per settore per un determinato concerto

Artista

Elodie

Date

2021-01-21 Roma

Vai!

	nomeSettore	prezzo	numLiberi
▶	palco	45	0
	platea	15	2

Figura 4.3: Interfaccia Utente