|  |  |
| --- | --- |
|  |  |

Basi di Dati e Conoscenza

Progetto A.A. 2018/2019

SISTEMA DI GESTIONE DI CORSI DI LINGUE STRANIERE

0242178

Antonio D’Orazio

**Indice**

[1. Descrizione del Minimondo 3](#_Toc606296459)

[2. Analisi dei Requisiti 4](#_Toc1289394997)

[3. Progettazione concettuale 9](#_Toc2081466291)

[4. Progettazione logica 13](#_Toc2147004904)

[5. Progettazione fisica 18](#_Toc518560220)

[Appendice: Implementazione 25](#_Toc403811585)

# Descrizione del Minimondo

|  |  |
| --- | --- |
|  |  |
| 1 2 3 4 5 6 7 8 9 101112131415161718192021222324252627282930313233 | Si progetti un sistema informativo per la gestione dei corsi di lingua inglese, tenuti presso un instituto di insegnamento. Tutte le informazioni fanno riferimento ad un solo anno scolastico in corso, e non viene richiesto di mantenere le informazioni relative agli anni scolastici precedenti (è quindi necessario prevedere un’opportuna funzionalità per indicare che si vuole riconfigurare il sistema per l’avvio di un nuovo anno scolastico). La base dati deve avere le seguenti caratteristiche e mantenere le seguenti informazioni.  I corsi sono organizzati per livelli. Ciascun livello è identificato dal nome del livello stesso (ad esempio Elementary, Intermediate, First Certificate, Advanced, Proficiency); inoltre è specificato il nome del libro di testo e se viene richiesto di sostenere un esame finale.  I corsi sono identificati univocamente dal nome del livello cui afferiscono e da un codice progressivo, necessario per distinguere corsi che fanno riferimento allo stesso livello. Per ciascun corso sono note la data di attivazione, il numero e le informazioni anagrafiche degli iscritti e l’elenco dei giorni ed orari in cui è tenuto.  Per gli insegnanti sono noti il nome, l’indirizzo, la nazione di provenienza, ed i corsi a cui sono stati assegnati. Ad un corso può essere assegnato più di un insegnante, assicurandosi che in una determinata fascia oraria un insegnante sia assegnato ad un solo corso.  Per gli allievi sono noti il nome, un recapito, il corso a cui sono iscritti, la data di iscrizione al corso e il numero di assenze fatte finora (è di interesse tenere traccia dei giorni specifici in cui un allievo è stato assente). Gli allievi possono anche prenotare lezioni private, qualora vogliano approfondire alcuni aspetti della lingua inglese. Si vuole tener traccia di tutte le lezioni private eventualmente richieste da un allievo, in quale data e con quale insegnante. La prenotazione di una lezione individuale non può avvenire in concomitanza di un altro impegno di un insegnante.  La scuola organizza anche un insieme di attività culturali. Ciascuna attività è identificata da un codice progressivo, e sono noti il giorno e l’ora in cui verrà tenuta. Nel caso di proiezioni in lingua originale, sono noti il nome del film ed il nome del regista. Nel caso di conferenze, sono noti l’argomento che verrà trattato ed il nome del conferenziere. Per poter partecipare alle attività gli allievi devono iscriversi.  Il personale amministrativo della scuola deve poter inserire all’interno del sistema informativo tutte le informazioni legate ai corsi ed agli insegnanti e possono generare dei report indicanti, su base mensile, quali attività hanno svolto gli insegnanti. Il personale di segreteria gestisce le iscrizioni degli utenti della scuola ai corsi. Gli insegnanti possono generare dei report indicanti la propria agenda, su base settimanale. |

# Analisi dei Requisiti

## Identificazione dei termini ambigui e correzioni possibili

|  |  |  |  |
| --- | --- | --- | --- |
| **Linea** | **Termine** | **Nuovo termine** | **Motivo correzione** |
| 9 | Nome | Titolo | ‘Nome’ è un termine utilizzato in più contesti. |
| 7, 10 | Nome | Denominazione | ‘Nome’ è un termine utilizzato in più contesti. |
| 12 | Informazioni Anagrafiche | Nome, cognome, sesso, data di nascita, luogo di nascita | Ambiguità su ciò che è richiesto. Suddivisione del termine in attributi atomici. |
| 17 | Recapito | Indirizzo | Sinonimi. Il termine ‘Indirizzo’ era già stato utilizzato in precedenza. |
| 14, 17, 26, 27 | Nome | Nome e cognome | Il termine ‘Nome’ in questo caso crea ambiguità poiché si riferisce ad un nominativo completo (Nome e cognome). Ciò è motivato dal fatto che è illogico salvare solamente il nome e non anche il cognome. Discorso analogo per le altre entità coinvolte che utilizzano questo attributo. |
| 28 | Attività | Attività culturali | Per non confondersi con attività inteso come impegno. |
| 13 | Iscritti | Allievi | Sinonimo di allievi. |
| 31 | Attività | Impegni | Sinonimo di impegno, per non confondersi con Attività Culturale. |
| 17 | Recapito | Indirizzo | Sinonimo di indirizzo. |
| 32 | Utenti della scuola | Allievi | Sinonimo di allievi. |
| 22 | Lezione individuale | Lezione private | Sinonimo. |
| 33 | Agenda | Lista degli impegni | Per rendere più chiaro il legame tra l’agenda e gli impegni. |

### Specifica disambiguata

|  |
| --- |
| Si progetti un sistema informativo per la gestione dei corsi di lingua inglese, tenuti presso un istituto di insegnamento. Tutte le informazioni fanno riferimento ad un solo anno scolastico in corso, e non viene richiesto di mantenere le informazioni relative agli anni scolastici precedenti (è quindi necessario prevedere un’opportuna funzionalità per indicare che si vuole riconfigurare il sistema per l’avvio di un nuovo anno scolastico). La base dati deve avere le seguenti caratteristiche e mantenere le seguenti informazioni.  I corsi sono organizzati per livelli. Ciascun livello è identificato dalla denominazione del livello stesso (ad esempio Elementary, Intermediate, First Certificate, Advanced, Proficiency); inoltre è specificato il titolo del libro di testo e se viene richiesto di sostenere un esame finale.  I corsi sono identificati univocamente dalla denominazione del livello cui afferiscono e da un codice progressivo, necessario per distinguere corsi che fanno riferimento allo stesso livello. Per ciascun corso sono note la data di attivazione, il numero degli allievi iscritti, nome, cognome, sesso, data di nascita, luogo di nascita degli allievi e l’elenco dei giorni ed orari in cui è tenuto.  Per gli insegnanti sono noti il nome, il cognome, l’indirizzo, la nazione di provenienza, ed i corsi a cui sono stati assegnati. Ad un corso può essere assegnato più di un insegnante, assicurandosi che in una determinata fascia oraria un insegnante sia assegnato ad un solo corso.  Per gli allievi sono noti il nome, il cognome, il sesso, la data di nascita, il luogo di nascita, l’indirizzo, il corso a cui sono iscritti, la data di iscrizione al corso e il numero di assenze fatte finora (è di interesse tenere traccia dei giorni specifici in cui un allievo è stato assente). Gli allievi possono anche prenotare lezioni private, qualora vogliano approfondire alcuni aspetti della lingua inglese. Si vuole tener traccia di tutte le lezioni private eventualmente richieste da un allievo, in quale data e con quale insegnante. La prenotazione di una lezione privata non può avvenire in concomitanza di un altro impegno di un insegnante.  La scuola organizza anche un insieme di attività culturali. Ciascuna attività è identificata da un codice progressivo, e sono noti il giorno e l’ora in cui verrà tenuta. Nel caso di proiezioni in lingua originale, sono noti il titolo del film, con nome e cognome del regista. Nel caso di conferenze, sono noti l’argomento che verrà trattato con nome e cognome del conferenziere. Per poter partecipare alle attività culturali gli allievi devono iscriversi.  Il personale amministrativo della scuola deve poter inserire all’interno del sistema informativo tutte le informazioni legate ai corsi ed agli insegnanti e possono generare dei report indicanti, su base mensile, quali impegni hanno svolto gli insegnanti. Il personale di segreteria gestisce le iscrizioni degli allievi ai corsi. Gli insegnanti possono generare dei report indicanti la propria lista degli impegni, su base settimanale. |

## Glossario dei Termini

|  |  |  |  |
| --- | --- | --- | --- |
| **Termine** | **Descrizione** | **Sinonimi** | **Collegamenti** |
| Corso | Corsi offerti. Si fa riferimento al solo anno scolastico attuale |  | Livello, Allievo, Insegnante |
| Livello | Rappresenta i vari livelli di inglese insegnati |  | Corso |
| Insegnante | Gli insegnanti dei corsi |  | Corso, Lezione privata |
| Lezione privata | Lezione tenuta privatamente da un insegnante verso un allievo | Lezione individuale | Corso, Allievo |
| Allievo | Coloro che partecipano ai corsi | Iscritto | Lezione private, Corso |
| Attività Culturale | Le attività culturali al di fuori delle lezioni organizzate dalla Scuola | Attività | Allievo |

## Raggruppamento dei requisiti in insiemi omogenei

|  |
| --- |
| **Frasi di carattere generale** |
| Si progetti un sistema informativo per la gestione dei corsi di lingua inglese, tenuti presso un istituto di insegnamento. Tutte le informazioni fanno riferimento ad un solo anno scolastico in corso, e non viene richiesto di mantenere le informazioni relative agli anni scolastici precedenti (è quindi necessario prevedere un’opportuna funzionalità per indicare che si vuole riconfigurare il sistema per l’avvio di un nuovo anno scolastico). |

|  |
| --- |
| **Frasi relative ai Corsi** |
| I corsi sono organizzati per livelli.  I corsi sono identificati univocamente dalla denominazione del livello cui afferiscono e da un codice progressivo, necessario per distinguere corsi che fanno riferimento allo stesso livello. Per ciascun corso sono note la data di attivazione, il numero degli allievi iscritti, nome, cognome, sesso, data di nascita, luogo di nascita degli allievi e l’elenco dei giorni ed orari in cui è tenuto. |

|  |
| --- |
| **Frasi relative ai Livelli** |
| Ciascun livello è identificato dalla denominazione del livello stesso (ad esempio Elementary, Intermediate, First Certificate, Advanced, Proficiency); inoltre è specificato il titolo del libro di testo e se viene richiesto di sostenere un esame finale. |

|  |
| --- |
| **Frasi relative agli Insegnanti** |
| Per gli insegnanti sono noti il nome, il cognome, l’indirizzo, la nazione di provenienza, ed i corsi a cui sono stati assegnati. Ad un corso può essere assegnato più di un insegnante, assicurandosi che in una determinata fascia oraria un insegnante sia assegnato ad un solo corso.  Gli insegnanti possono generare dei report indicanti la propria lista degli impegni, su base settimanale. |

|  |
| --- |
| **Frasi relative agli Allievi** |
| Per gli allievi sono noti il nome, il cognome, il sesso, la data di nascita, il luogo di nascita, l’indirizzo, il corso a cui sono iscritti, la data di iscrizione al corso e il numero di assenze fatte finora (è di interesse tenere traccia dei giorni specifici in cui un allievo è stato assente). Gli allievi possono anche prenotare lezioni private, qualora vogliano approfondire alcuni aspetti della lingua inglese. |

|  |
| --- |
| **Frasi relative alle Lezioni Private** |
| Si vuole tener traccia di tutte le lezioni private eventualmente richieste da un allievo, in quale data e con quale insegnante. La prenotazione di una lezione privata non può avvenire in concomitanza di un altro impegno di un insegnante. |

|  |
| --- |
| **Frasi relative alle Attività Culturali** |
| La scuola organizza anche un insieme di attività culturali. Ciascuna attività è identificata da un codice progressivo, e sono noti il giorno e l’ora in cui verrà tenuta. Nel caso di proiezioni in lingua originale, sono noti il titolo del film, con nome e cognome del regista. Nel caso di conferenze, sono noti l’argomento che verrà trattato con nome e cognome del conferenziere. Per poter partecipare alle attività culturali gli allievi devono iscriversi. |

# Progettazione concettuale

## Costruzione dello schema E-R

Per prima cosa individuo, sulla base degli insiemi omogenei rilevati, le entità candidate a comporre lo schema finale.

Lezioni Private

Allievi

Insegnanti

Livelli

Corsi

Attività Culturali

Noto che una attività culturale può rappresentare in maniera esclusiva una proiezione in lingua o una conferenza. Pertanto, eseguo un primo raffinamento costruendo una generalizzazione.

Attività Culturali

Conferenze

Proiezioni

Noto inoltre che un corso è costituito da diverse lezioni, le quali indicano data, ora ed aula in cui sarà svolta. Aggiungo quindi una ulteriore entità.

Lezioni

Questa entità sarà anche messa in relazione con l’entità Allievi per identificare le assenze, il quale numero, così come quello degli allievi iscritti ad un corso, sarà ricavato tramite conteggio.

A questo punto raffino l’intero schema definendo le relazioni, sulla base dei collegamenti individuati in precedenza, seguendo la metodologia top down

Lezioni

Composizione

Afferenza

Livelli

Corsi

Iscrizione

Assenza

Docenza

Prenotazione

Allievi

Insegnanti

Frequenta

Attività Culturali

Conferenze

Proiezioni

Impartisce

Lezioni Private

### Integrazione finale

Procedo indicando gli attributi, gli identificatori e le molteplicità delle relazioni. Avendo proceduto con l’integrazione fra le diverse entità sin da subito, non sono stati riscontrati conflitti.

Aula

Ora

Giorno

Corsi

Livelli

Insegnanti

Allievi

Lezioni Private

Attività Culturali

Conferenze

Proiezioni

Afferenza

Prenotazione

Frequenta

Docenza

Impartisce

Iscrizione

Assenza

Lezioni

Composizione

(1, 1)

(1, n)

Denominazione

Codice

(1, n)

(1, 1)

Data Attivazione

Titolo

Esame necessario

(1, n)

(1, n)

(0, n)

(1, 1)

Data Iscrizione

(0, n)

(1, n)

(0, n)

Cognome

Pwd

Nome

Indirizzo

Cognome

Nome

Pwd

Data Nascita

Sesso

(0, n)

(0, n)

CF

Codice

Luogo nascita

Giorno

Indirizzo

Ora

Nazione

CF

(1, 1)

(0, n)

(1, 1)

Giorno

Ora

Nome Regista

Nome Regista

Cognome Regista

Cognome Regista

Titolo Film

Titolo Film

## Regole aziendali

- Una lezione privata non deve essere prenotata in un orario in cui l’insegnante è già impegnato

- Una assenza non deve essere aggiunta in una lezione ancora da verificarsi

- Non deve essere possibile aggiungere una assenza se l’allievo non appartiene al corso della data lezione

- In una determinata fascia oraria un insegnante deve essere assegnato solo ad un corso

- In una determinata fascia oraria un’aula deve ospitare una ed una sola lezione

## Dizionario dei dati

|  |  |  |  |
| --- | --- | --- | --- |
| **Entità** | **Descrizione** | **Attributi** | **Identificatori** |
| Corsi | Rappresenta i corsi attivati per l’anno accademico corrente | Codice, Data attivazione | Codice |
| Lezioni | Rappresenta le lezioni per ogni corso attivato, indicandone ora, aula e giorno | Aula, giorno, ora | Aula, giorno, ora |
| Insegnanti | Rappresenta gli insegnanti impiegati nella scuola, indicandone anche la nazione di provenienza. Il campo password sarà utilizzato per visualizzare l’agenda per ogni insegnante | CF, pwd, cognome, nome, indirizzo, nazione | CF |
| Livelli | Rappresentano i livelli dei corsi insegnati nella scuola | Denominazione, Titolo Libro, Esame Necessario | Denominazione |
| Lezioni Private | Rappresenta gli appuntamenti tra insegnanti ed allievi per le lezioni private | Insegnante, Allievo, Data, Ora, Aula | Allievo, Data, Ora, Aula |
| Allievi | Rappresenta gli allievi iscritti alla scuola. Ogni allievo è iscritto ad un singolo corso | CF, Cognome, Nome, Indirizzo, Sesso, Data Nascita, Luogo Nascita Password | CF |
| Attività Culturali | Rappresenta le attività culturali attivate nella scuola | Codice, Giorno, Ora | Codice |
| Proiezioni | Rappresenta in dettaglio le proiezioni in lingua, esse sono attività culturali | Titolo film, cognome regista, nome regista | Codice (da Attivita Culturali) |
| Conferenze | Rappresenta in dettaglio le conferenze. Sono attività culturali anch’esse. | Argomento, Nome conferenziere, cognome conferenziere | Codice (da Attivita Culturali) |

# Progettazione logica

## Volume dei dati

|  |  |  |
| --- | --- | --- |
| **Concetto nello schema** | **Tipo[[1]](#footnote-2)** | **Volume atteso** |
| Corsi | E | 15 |
| Lezioni | E | 1485 |
| Insegnanti | E | 30 |
| Livelli | E | 5 |
| Allievi | E | 450 |
| Lezioni private | E | 165 |
| Attività Culturali | E | 66 |
| Proiezioni | E | 33 |
| Conferenze | E | 33 |
| Composizione | R | 1485 |
| Docenza | R | 30 |
| Impartisce | R | 165 |
| Afferenza | R | 15 |
| Assenze | R | 5569 |
| Frequenta | R | 200 |
| Iscrizione | R | 450 |
| Prenotazione | R | 3300 |

## Tavola delle operazioni

|  |  |  |
| --- | --- | --- |
| **Cod.** | **Descrizione** | **Frequenza attesa** |
| 1 | Attivazione di un corso | 15 in un anno |
| 2 | Iscrizione di un nuovo allievo ad un corso | 450 in un anno |
| 3 | Assegnazione di un insegnante ad un corso | 30 in un anno |
| 4 | Prenotazione di una lezione privata | 5 a settimana |
| 5 | Visualizzazione delle lezioni private per ogni allievo | 250 al giorno |
| 6 | Attivazione di una attività culturale | 2 a settimana |
| 7 | Prenotazione ad una attività culturale | 100 a settimana |
| 8 | Visualizzare per ogni allievo le sue prenotazioni alle attività culturali future | 80 a settimana |
| 9 | Report su base mensile della segreteria | 1 al mese |
| 10 | Report su base settimanale dell’insegnante | 30 a settimana |
| 11 | Visualizzazione della lista di assenze per ogni allievo | 200 a settimana |
| 12 | Aggiunta di una nuova assenza da parte dell’insegnante | 169 a settimana |
| 13 | Inizializzazione di un nuovo anno | 1 all’anno |
| 14 | Visualizzazione dei corsi attivi con il rispettivo numero di alunni iscritti | 30 in un anno |
| 15 | Aggiunta di un insegnante al sistema | 30 in un anno |
| 16 | Aggiunta di una lezione ad un corso | 1485 in un anno |

## Costo delle operazioni

Operazione 1:

Una scrittura su Corsi. Costo 2

Costo finale: 30 accessi in un anno.

Operazione 2:

Una scrittura su Allievi. Costo 2.

Una scrittura su Iscrizione. Costo 2.

Costo complessivo: 4.

Costo finale: 3800 accessi in un anno.

Operazione 3:

Una scrittura su Docenza. Costo 2.

Costo finale: 60 accessi in un anno.

Operazione 4:

Una scrittura su Lezioni Private. Costo 2.

Costo finale: 10 accessi a settimana.

Operazione 5:

Una lettura su Lezioni Private. Costo 1.

Costo finale: 250 accessi al giorno

Operazione 6:

Una scrittura su Attività Culturale. Costo 2.

Una scrittura su Proiezioni oppure su Conferenze. Costo 2.

Costo complessivo: 4.

Costo finale: 8 accessi a settimana

Operazione 7:

Una scrittura su Prenotazione. Costo 2.

Costo finale: 200 accessi a settimana

Operazione 8:

Si suppone che le prenotazioni siano per la settimana corrente

Due letture su Prenotazione. Costo 2.

Costo finale: 160 accessi a settimana

Operazione 9:

Si suppone che un insegnante tenga in media una lezione privata a settimana e che in ogni settimana ci siano tre lezioni per corso.

30 Letture su Insegnanti. Costo 30

Per ogni insegnante:

Quattro letture su Lezioni Private. Costo 4

Una lettura su Docenze. Costo 1

Una lettura su Corsi. Costo 1.

12 letture su Lezioni. Costo 12.

Costo complessivo: 480

Costo finale: 480 accessi al mese

Operazione 10:

Una lettura su Insegnanti. Costo 1.

Una lettura su Lezioni Private. Costo 1.

Una lettura su Docenze. Costo 1

Una lettura su Corsi. Costo 1.

Tre letture su Lezioni. Costo 3.

Costo complessivo: 7

Costo finale: 210 accessi a settimana.

Operazione 11:

Nella ipotesi che durante l’anno sia stata già compiuta la media di 5 assenze per allievo.

Cinque letture su Assenze. Costo 5

Costo complessivo: 5

Costo finale: 1000 accessi a settimana

Operazione 12:

Una scrittura su Assenze. Costo 2.

Costo finale: 338 accessi a settimana.

Operazione 13:

15 Cancellazioni su Corsi

1485 cancellazioni su Lezioni

450 cancellazioni su Allievi

165 cancellazioni su Lezioni Private

66 cancellazioni su Attività Culturali

33 cancellazioni su Proiezioni

33 cancellazioni su Conferenze

1485 cancellazioni su Composizione

30 cancellazioni su Docenza

165 cancellazioni su Impartisce

15 cancellazioni su Afferenza

5569 cancellazioni su Assenze

200 cancellazioni su Frequenta

450 cancellazioni su Iscrizione

3300 cancellazioni su Prenotazione

Costo totale: 13.461 accessi in un anno

Operazione 14:

15 Letture su Corsi

15 Letture su Afferenza

5 Letture su Livelli

450 Letture su Iscrizioni

450 Letture su Allievi

Costo complessivo: 935

Costo totale: 28.050 accessi in un anno

Operazione 15:

Nella peggiore ipotesi cambiano tutti gli insegnanti

Un inserimento su Insegnanti

Costo complessivo: 60

Costo totale: 60 accessi in un anno

Operazione 16:

Un inserimento su Lezioni

Costo complessivo: 2970

Costo totale: 2970 accessi in un anno

## Ristrutturazione dello schema E-R

Per ridurre il costo nella prenotazione delle attività culturali, l’operazione più onerosa, scelgo di eliminare la generalizzazione creando due entità separate. In questo modo gli allievi potranno a monte scegliere il tipo di attività per la quale prenotarsi eliminando così la visualizzazione di tipi diversi. Inoltre, è utile e soprattutto anche ai fini dell’operazione 8, dimezzando il carico delle relazioni di Prenotazione, sempre selezionando a monte la categoria su cui effettuare l’operazione.

La ristrutturazione avverrà quindi eliminando la generalizzazione in questo modo:

Allievi

(0, n)

(0, n)

Prenotazione

Proiezione

Prenotazione

Conferenza

Codice

Giorno

Ora

Cognome Conferenziere

Titolo Film

Nome Conferenziere

Conferenze

Cognome Regista

Nome Regista

Titolo Film

Proiezioni

Giorno

Ora

Codice

(0, n)

(0, n)

## Trasformazione di attributi e identificatori

È presente un identificatore esterno in Lezioni Private. La trasformazione eseguita è quella di aggiungere un nuovo attributo “Allievo” al posto dell’identificatore esterno, con vincolo di integrità referenziale tra CF in Allievo e Allievo in Lezioni Private.

Da questa porzione iniziale di modello:

Impartisce

Allievi

Lezioni Private

Frequenta

Cognome

Nome

Pwd

Insegnanti

Indirizzo

CF

Nazione

Ora

Giorno

CF

Pwd

Cognome

Nome

(0, n)

(1, 1)

(1, 1)

(0, n)

Indirizzo

Sesso

Data Nascita

Luogo nascita

Si ottiene, ristrutturando:

Impartisce

Allievi

Lezioni Private

Frequenta

Cognome

Nome

Pwd

Insegnanti

Indirizzo

CF

Nazione

Ora

Giorno

CF

Pwd

Cognome

Nome

(0, n)

(1, 1)

(1, 1)

(0, n)

Allievo

Indirizzo

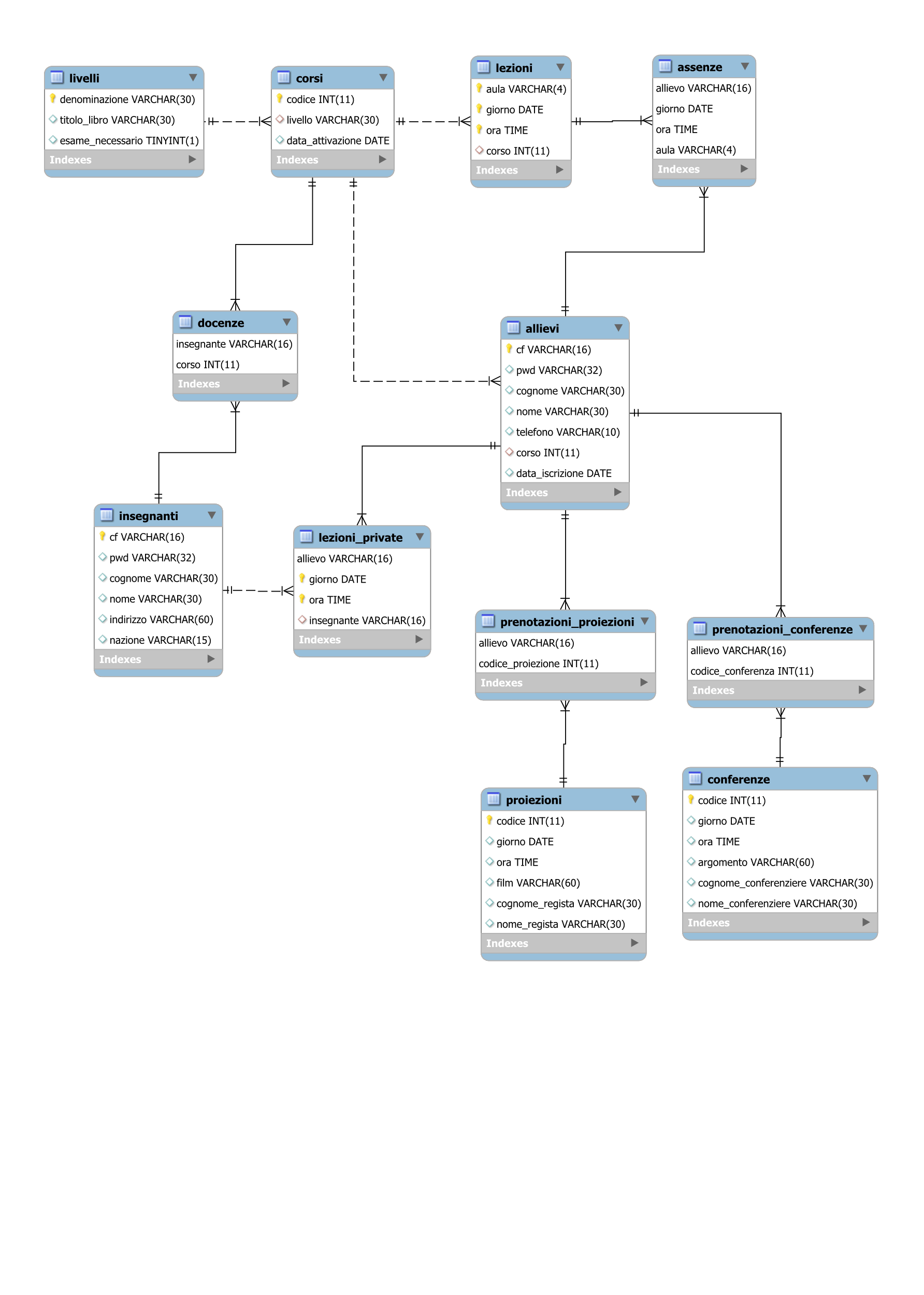
Sesso

Data Nascita

Luogo nascita

## Traduzione di entità e associazioni

Note: la linea continua indica che un’entità debole (es. docenze) utilizza come primary key (o in parte) delle foreign key che fanno riferimento ad altre tabelle. Questi campi si riconoscono in quanto non hanno alcun simbolo esplicito a fianco.



## Normalizzazione del modello relazionale

Il modello relazionale è già in 1NF in quanto per ogni relazione esiste la relativa chiave primaria e tutti gli attributi rappresentano un valore atomico.

Il modello è anche già in 2NF, in quanto per ogni relazione si ha che ogni attributo non chiave dipende completamente da ogni chiave, e non solamente da una parte di essa.

Per quanto riguarda la terza forma normale, ci sono delle violazioni.

Proiezioni(Codice, Giorno, Ora, TitoloFilm, CognomeRegista, NomeRegista)

Conferenze(Codice, Giorno, Ora, Argomento, NomeConferenziere, CognomeConferenziere)

Poiché CognomeRegista e Nome regista dipendono funzionalmente da TitoloFilm, e quindi vi è una relazione di dipendenza funzionale transitiva rispetto alla chiave primaria.

Una decomposizione in 3NF è la seguente:

Proiezioni(Codice, Giorno, Ora, Film)

Film(Id\_Film, TitoloFilm, regista);

Registi(id\_regista, cognome, nome);

Dove:

Proiezione identifica l’evento della scuola

Film identifica la pellicola effettiva da proiettare

Registi identifica il regista del film

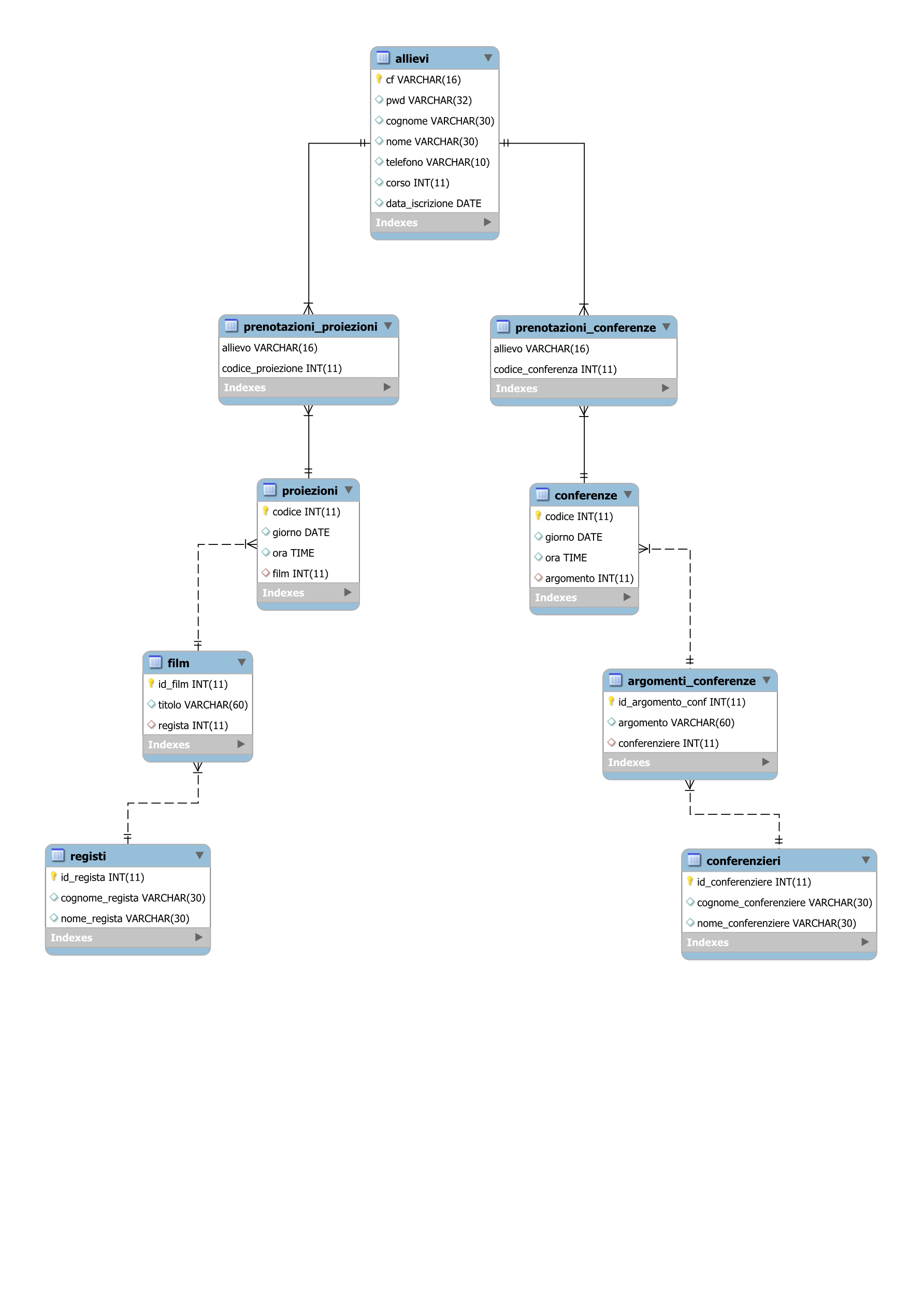
Analogamente per Conferenze

Conferenze(Codice, Giorno, Ora, Argomento);

ArgomentiConferenze(id\_argomento, Argomento, conferenziere)

Conferenzieri(id\_conferenziere, nome, cognome)

In forma grafica:



# Progettazione fisica

## Utenti e privilegi

Sono previste tre utenze, indirizzate alle tre categorie di utenti che utilizzeranno l’applicazione. Allievi ed insegnanti avranno permessi specifici in lettura e/o scrittura in base alle tabelle, viste e procedure che dovranno utilizzare, mentre alla segreteria è garantito l’accesso generale. Allievi ed insegnanti potranno poi autenticarsi tramite CF e Password come memorizzati nelle rispettive tabelle, tramite il portale.

*user-allievi:*

Può leggere: Allievi, ProiezioniCompleta, ConferenzeCompleta, PrenotazioniAttive, Assenze, Prenotazioni\_Lezioni\_Private, Insegnanti

Può leggere e scrivere su: Prenotazioni\_Conferenze, Prenotazioni\_Proiezioni, Lezioni\_Private

Può eseguire: Prenotazione\_Attività

*user-insegnanti:*

Può leggere: Lezioni, Insegnanti, Impegni\_Insegnante, Lezioni\_Private, Docenze, Corsi

Può leggere e scrivere su: Assenze

*user-segreteria:*

Tutti i permessi, per le operazioni di gestione e di manutenzione e per resettare l’anno scolastico.

## Strutture di memorizzazione

|  |  |  |
| --- | --- | --- |
| **Tabella Livelli** | | |
| **Attributo** | **Tipo di dato** | **Attributi[[2]](#footnote-3)** |
| **denominazione** | VARCHAR(30) | PK |
| **titolo\_libro** | VARCHAR(30) |  |
| **esame\_necessario** | BOOLEAN |  |

|  |  |  |
| --- | --- | --- |
| **Tabella Corsi** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **codice** | INTEGER | PK, AI |
| **livello** | VARCHAR(30) |  |
| **data\_attivazione** | DATE |  |

|  |  |  |
| --- | --- | --- |
| **Tabella Insegnanti** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **cf** | VARCHAR(16) | PK |
| **pwd** | VARCHAR(32) |  |
| **cognome** | VARCHAR(30) |  |
| **nome** | VARCHAR(30) |  |
| **indirizzo** | VARCHAR(60) |  |
| **nazione** | VARCHAR(15) |  |

|  |  |  |
| --- | --- | --- |
| **Tabella Allievi** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **cf** | VARCHAR(16) | PK |
| **pwd** | VARCHAR(32) |  |
| **cognome** | VARCHAR(30) |  |
| **nome** | VARCHAR(30) |  |
| **sesso** | ENUM(‘M’, ‘F’) |  |
| **data\_nascita** | DATE |  |
| **luogo\_nascita** | VARCHAR(15) |  |
| **indirizzo** | VARCHAR(60) |  |
| **corso** | INTEGER |  |
| **data\_iscrizione** | DATE |  |

|  |  |  |
| --- | --- | --- |
| **Tabella Lezioni** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **aula** | VARCHAR(4) | PK |
| **giorno** | DATE | PK |
| **ora** | TIME | PK |
| **corso** | INTEGER |  |

|  |  |  |
| --- | --- | --- |
| **Tabella Lezioni\_Private** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **allievo** | VARCHAR(16) | PK |
| **giorno** | DATE | PK |
| **ora** | TIME | PK |
| **insegnante** | VARCHAR(16) |  |

|  |  |  |
| --- | --- | --- |
| **Tabella Docenze** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **insegnante** | VARCHAR(16) | PK |
| **corso** | INTEGER | PK |

|  |  |  |
| --- | --- | --- |
| **Tabella Assenze** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **allievo** | VARCHAR(16) | PK |
| **giorno** | DATE | PK |
| **ora** | TIME | PK |
| **aula** | VARCHAR(4) | PK |

|  |  |  |
| --- | --- | --- |
| **Tabella Registi** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **id\_regista** | INTEGER | PK, AI |
| **cognome\_regista** | VARCHAR(30) |  |
| **nome\_regista** | VARCHAR(30) |  |

|  |  |  |
| --- | --- | --- |
| **Tabella Film** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **id\_film** | INTEGER | PK, AI |
| **titolo** | VARCHAR(60) |  |
| **regista** | INTEGER |  |

|  |  |  |
| --- | --- | --- |
| **Tabella Argomenti\_Conferenze** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **id\_argomento\_conf** | INTEGER | PK |
| **argomento** | VARCHAR(60) |  |
| **conferenziere** | INTEGER |  |

|  |  |  |
| --- | --- | --- |
| **Tabella Proiezioni** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **codice** | INTEGER | PK, AI |
| **giorno** | DATE |  |
| **ora** | TIME |  |
| **film** | INTEGER |  |

|  |  |  |
| --- | --- | --- |
| **Tabella Conferenzieri** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **id\_conferenziere** | INTEGER | PK, AI |
| **cognome\_conferenziere** | VARCHAR(30) |  |
| **nome\_conferenziere** | VARCHAR(30) |  |

|  |  |  |
| --- | --- | --- |
| **Tabella Conferenze** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **codice** | INTEGER | PK, AI |
| **giorno** | DATE |  |
| **ora** | TIME |  |
| **argomento** | INTEGER |  |

|  |  |  |
| --- | --- | --- |
| **Tabella Prenotazioni\_Proiezioni** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **allievo** | VARCHAR(16) | PK |
| **codice\_proiezione** | INTEGER | PK |

|  |  |  |
| --- | --- | --- |
| **Tabella Prenotazioni\_Conferenze** | | |
| **Attributo** | **Tipo di dato** | **Attributi** |
| **allievo** | VARCHAR(16) | PK |
| **codice\_conferenza** | INTEGER | PK |

## Indici

I seguenti indici sono generati automaticamente dal DBMS in quanto appartenenti a chiavi primarie ed esterne.

Ho creato inoltre gli indici proiezioni\_idx e conferenze\_idx che indicizzano i campi (giorno, ora) rispettivamente delle tabelle Proiezioni e Conferenze, consentendo così di velocizzare la ricerca di un evento per data.

|  |  |
| --- | --- |
| **Tabella Allievi** | |
| **Indice PRIMARY** | **Tipo[[3]](#footnote-4): PR** |
| Colonna 1 | cf |
| **Indice corso** | **Tipo: IDX** |
| Colonna 1 | corso |

|  |  |
| --- | --- |
| **Tabella Argomenti\_Conferenze** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | id\_argomento\_conf |
| **Indice conferenziere** | **Tipo: IDX** |
| Colonna 1 | conferenziere |

|  |  |
| --- | --- |
| **Tabella Assenze** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | allievo |
| Colonna 2 | giorno |
| Colonna 3 | ora |
| Colonna 4 | aula |
| **Indice aula** | **Tipo: IDX** |
| Colonna 1 | aula |
| Colonna 2 | giorno |
| Colonna 3 | ora |

|  |  |
| --- | --- |
| **Tabella Conferenze** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | codice |
| **Indice argomento** | **Tipo: IDX** |
| Colonna 1 | argomento |
| **Indice conferenze\_idx** | **Tipo: IDX** |
| Colonna 1 | giorno |
| Colonna 2 | ora |

|  |  |
| --- | --- |
| **Tabella Conferenzieri** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | id\_conferenziere |

|  |  |
| --- | --- |
| **Tabella Corsi** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | codice |
| **Indice livello** | **Tipo: IDX** |
| Colonna 1 | livello |

|  |  |
| --- | --- |
| **Tabella Docenze** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | insegnante |
| Colonna 2 | corso |
| **Indice corso** | **Tipo: IDX** |
| Colonna 1 | corso |

|  |  |
| --- | --- |
| **Tabella Film** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | id\_film |
| **Indice regista** | **Tipo: IDX** |
| Colonna 1 | regista |

|  |  |
| --- | --- |
| **Tabella Insegnanti** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | cf |

|  |  |
| --- | --- |
| **Tabella Lezioni** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | aula |
| Colonna 2 | giorno |
| Colonna 3 | ora |
| **Indice corso** | **Tipo: IDX** |
| Colonna 1 | corso |

|  |  |
| --- | --- |
| **Tabella Lezioni\_Private** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | allievo |
| Colonna 2 | giorno |
| Colonna 3 | ora |
| **Indice insegnante** | **Tipo: IDX** |
| Colonna 1 | insegnante |

|  |  |
| --- | --- |
| **Tabella Livelli** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | denominazione |

|  |  |
| --- | --- |
| **Tabella Prenotazioni\_Conferenze** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | allievo |
| Colonna 2 | codice\_conferenza |
| **Indice codice\_conferenza** | **Tipo: IDX** |
| Colonna 1 | codice\_conferenza |

|  |  |
| --- | --- |
| **Tabella Prenotazioni\_Proiezioni** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | allievo |
| Colonna 2 | codice\_proiezione |
| **Indice codice\_proiezione** | **Tipo: IDX** |
| Colonna 1 | codice\_proiezione |

|  |  |
| --- | --- |
| **Tabella Proiezioni** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | codice |
| **Indice film** | **Tipo: IDX** |
| Colonna 1 | film |
| **Indice proiezioni\_idx** | **Tipo: IDX** |
| Colonna 1 | giorno |
| Colonna 2 | ora |

|  |  |
| --- | --- |
| **Tabella Registi** | |
| **Indice PRIMARY** | **Tipo: PR** |
| Colonna 1 | id\_regista |

## Trigger

DELIMITER $$

CREATE TRIGGER controllo\_orario\_corsi\_trg

BEFORE INSERT

ON Docenze FOR EACH ROW

BEGIN

IF EXISTS

(

SELECT l.giorno, l.ora

FROM Lezioni l JOIN Corsi c

ON c.codice = l.corso

JOIN Docenze d ON d.corso = c.codice

WHERE d.insegnante = NEW.insegnante

AND c.codice <> NEW.corso

INTERSECT

SELECT l.giorno, l.ora

FROM Lezioni l

WHERE l.corso = NEW.corso

)

THEN

SIGNAL SQLSTATE '12345'

SET MESSAGE\_TEXT = 'Una o piu fasce orarie non disponibili!';

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE TRIGGER controllo\_orario\_lezione\_privata\_trg

BEFORE INSERT

ON Lezioni\_Private FOR EACH ROW

BEGIN

IF EXISTS

(

(SELECT i.cf AS CF\_Insegnante, i.cognome AS Cognome, i.nome AS Nome, lp.giorno

     AS Giorno, lp.ora AS Ora

FROM Insegnanti i JOIN Lezioni\_Private lp ON i.cf = lp.insegnante

WHERE i.cf = NEW.insegnante

AND HOUR(NEW.ora) = HOUR(lp.ora)

AND lp.giorno = NEW.giorno)

UNION

(SELECT i.cf AS CF\_Insegnante, i.cognome AS Cognome, i.nome AS Nome, l.giorno

     AS Giorno, l.ora AS Ora

FROM Insegnanti i JOIN Docenze d ON i.cf = d.insegnante

        JOIN Corsi c ON d.corso = c.codice

        JOIN Lezioni l ON l.corso = c.codice

WHERE i.cf = NEW.insegnante

AND HOUR(NEW.ora) = HOUR(l.ora)

AND l.giorno = NEW.giorno)

)

THEN

SIGNAL SQLSTATE '12345'

SET MESSAGE\_TEXT = 'Orario gia occupato!';

END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE TRIGGER controllo\_validita\_assenza\_trg

BEFORE INSERT

ON Assenze FOR EACH ROW

BEGIN

IF NOT EXISTS

(

SELECT \* FROM Lezioni l, Allievi al

WHERE l.aula = NEW.aula AND l.giorno = NEW.giorno AND l.ora = NEW.ora AND l.corso = al.corso AND al.cf = NEW.allievo

)

THEN

SIGNAL SQLSTATE '12345'

SET MESSAGE\_TEXT = 'Data o allievo non validi';

ELSEIF (NEW.giorno > NOW())

THEN

SIGNAL SQLSTATE '12345'

SET MESSAGE\_TEXT = 'Lezione ancora da verificarsi';

END IF;

END$$

DELIMITER ;

## Eventi

Nessun evento utilizzato.

## Viste

DROP VIEW IF EXISTS Impegni\_Insegnante;

-- Verra usata sia per la segreteria sia per il report per insegnante

CREATE VIEW Impegni\_Insegnante

AS

(SELECT i.cf AS CF\_Insegnante, i.cognome AS Cognome, i.nome AS Nome, lp.giorno

    AS Giorno, lp.ora AS Ora, lp.aula AS Aula, 'Lezione Privata' AS Tipologia

FROM insegnanti i JOIN lezioni\_private lp ON i.cf = lp.insegnante)

UNION

(SELECT i.cf AS CF\_Insegnante, i.cognome AS Cognome, i.nome AS Nome, l.giorno

    AS Giorno, l.ora AS Ora, l.aula AS Aula, 'Lezione Corso' AS Tipologia

FROM insegnanti i JOIN docenze d ON i.cf = d.insegnante

             JOIN corsi c ON d.corso = c.codice

             JOIN lezioni l ON l.corso = c.codice

)

ORDER BY Cognome, Nome, Giorno, Ora;

DROP VIEW IF EXISTS Prenotazioni\_Lezioni\_Private;

-- Verra usata dagli allievi per prenotare una lezione privata

CREATE VIEW Prenotazioni\_Lezioni\_Private

AS

SELECT a.cf AS CF\_Allievo, a.cogprenotazioni\_lezioni\_privatenome AS Cognome\_Allievo, a.nome AS Nome\_Allievo,

    i.cognome AS Cognome\_Insegnante, i.nome AS Nome\_Insegnante, lp.giorno

    AS Giorno, lp.ora AS Ora, lp.aula AS Aula

FROM Allievi a JOIN Lezioni\_Private lp ON lp.allievo = a.cf

    JOIN Insegnanti i ON lp.insegnante = i.cf

## Stored Procedures e transazioni

DELIMITER $$

CREATE PROCEDURE Nuova\_Attivita(IN TIPOLOGIA INTEGER, IN giorno\_in DATE, IN ora\_in TIME, IN argfilm\_in VARCHAR(60), IN cognome\_in VARCHAR(30), IN nome\_in VARCHAR(30))

BEGIN

    DECLARE idautore INTEGER;

DECLARE idopera INTEGER;

    -- Se è una proiezione

    IF TIPOLOGIA = 0 THEN

        -- Verifico esistenza regista sul db

SELECT id\_regista INTO idautore

FROM Registi

WHERE cognome\_regista = cognome\_in

AND nome\_regista = nome\_in;

IF idautore IS NULL THEN

            INSERT INTO Registi(cognome\_regista, nome\_regista)

VALUES (cognome\_in, nome\_in);

            SET idautore = LAST\_INSERT\_ID();

END IF;

        SELECT id\_film INTO idopera

        FROM Film

        WHERE titolo = argfilm\_in

        AND regista = idautore;

        -- Verifico esistenza film sul db

IF idopera IS NULL THEN

            INSERT INTO Film(titolo, regista)

VALUES (argfilm\_in, idautore);

SET idopera = LAST\_INSERT\_ID();

END IF;

INSERT INTO Proiezioni(giorno, ora, film)

VALUES (giorno\_in, ora\_in, idopera);

    -- Se è una conferenza TODO

    ELSEIF TIPOLOGIA = 1 THEN

SELECT id\_conferenziere INTO idautore

FROM Conferenzieri

WHERE cognome\_conferenziere = cognome\_in

AND nome\_conferenziere = nome\_in;

IF idautore IS NULL THEN

            INSERT INTO Conferenzieri(cognome\_conferenziere, nome\_conferenziere)

VALUES (cognome\_in, nome\_in);

            SET idautore = LAST\_INSERT\_ID();

END IF;

        SELECT id\_argomento\_conf INTO idopera

        FROM Argomenti\_Conferenze

        WHERE argomento = argfilm\_in

        AND conferenziere = idautore;

        -- Verifico esistenza film sul db

IF idopera IS NULL THEN

            INSERT INTO Argomenti\_Conferenze(argomento, conferenziere)

VALUES (argfilm\_in, idautore);

SET idopera = LAST\_INSERT\_ID();

END IF;

INSERT INTO Conferenze(giorno, ora, argomento)

VALUES (giorno\_in, ora\_in, idopera);

    ELSE

        ROLLBACK;

    END IF;

    COMMIT;

END$$

DELIMITER ;

DELIMITER $$

CREATE PROCEDURE PRENOTAZIONE\_ATTIVITA(IN TIPOLOGIA INTEGER, IN allievo\_in VARCHAR(16), IN codice\_in INTEGER)

BEGIN

    -- Se è una proiezione

    IF TIPOLOGIA = 0 THEN

        INSERT INTO Prenotazioni\_Proiezioni(allievo, codice\_proiezione)

        VALUES (allievo\_in, codice\_in);

    -- Se è una conferenza

    ELSEIF TIPOLOGIA = 1 THEN

        INSERT INTO Prenotazioni\_Conferenze(allievo, codice\_conferenza)

        VALUES (allievo\_in, codice\_in);

    ELSE

        ROLLBACK;

    END IF;

    COMMIT;

END$$

DELIMITER ;

DELIMITER $$

CREATE PROCEDURE Aggiungi\_Assenza(IN allievo\_in VARCHAR(16), IN giorno\_in DATE, IN ora\_in TIME, IN aula\_in VARCHAR(4), IN insegnante\_in VARCHAR(16))

BEGIN

-- Solo un insegnante abilitato al corso puo aggiungere una assenza

IF NOT EXISTS

(SELECT d.insegnante FROM Docenze d JOIN Corsi c ON d.corso = c.codice JOIN Lezioni l ON l.corso = c.codice

WHERE d.insegnante = insegnante\_in

AND l.giorno = giorno\_in

AND l.ora = ora\_in

AND l.aula = aula\_in)

THEN

SIGNAL SQLSTATE '12345'

SET MESSAGE\_TEXT = 'Insegnante non abilitato al corso';

ROLLBACK;

ELSE

INSERT INTO Assenze(allievo, giorno, ora, aula) VALUES (allievo\_in, giorno\_in, ora\_in, aula\_in);

     COMMIT;

    END IF;

END$$

DELIMITER ;

DELIMITER $$

CREATE PROCEDURE Nuovo\_Anno()

BEGIN

    TRUNCATE TABLE Iscrizioni;

    TRUNCATE TABLE Docenze;

    TRUNCATE TABLE Lezioni;

    TRUNCATE TABLE Assenze;

    TRUNCATE TABLE Prenotazioni\_Conferenze;

    TRUNCATE TABLE Prenotazioni\_Proiezioni;

    TRUNCATE TABLE Conferenze;

    TRUNCATE TABLE Proiezioni;

    TRUNCATE TABLE Corsi;

    COMMIT;

END$$

DELIMITER ;

# Appendice: Implementazione

## Codice SQL per instanziare il database

Riportare il codice SQL necessario ad istanziare lo schema del DB.

## Codice del Front-End

Riportare (correttamente formattato) il codice C del thin client realizzato per interagire con la base di dati.

1. Indicare con E le entità, con R le relazioni [↑](#footnote-ref-2)
2. PK = primary key, NN = not null, UQ = unique, UN = unsigned, AI = auto increment. È ovviamente possibile specificare più di un attributo per ciascuna colonna. [↑](#footnote-ref-3)
3. IDX = index, UQ = unique, FT = full text, PR = primary. [↑](#footnote-ref-4)