Università degli Studi di Napoli Federico II Scuola Politecnica e delle Scienze di Base

Dipartimento di Ingegneria Elettrica e Tecnologie dell’Informazione

Corso di Laurea in Informatica - Insegnamento di Basi di Dati I

Anno Accademico 2022/2023

Progettazione e Sviluppo di una Base di Dati Relazionale per la Gestione di un Laboratorio Scientifico

Antonio Abbatiello - N86003037

Leonardo Colamarino - N86003586

**??/??**/2023

Indice

1 Descrizione del Progetto5

1.1 Analisi del problema5

2 Progettazione Concettuale6

2.1 Introduzione6

2.2 Convenzione nell’uso di UML6

2.3 Class Diagram7

2.4 Ristrutturazione8

2.4.1 Analisi delle ridondanze8

2.4.2 Analisi degli identificativi8

2.4.3 Rimozione degli attributi multipli8

2.4.4 Rimozione delle classi di associazione9

2.4.5 Rimozione delle gerarchie di specializzazione9

2.4.6 Rimozione delle composizioni ed aggregazioni9

2.5 Class Diagram Ristrutturato10

2.6 Dizionario delle classi11

2.7 Dizionario delle associazioni14

2.8 Dizionario dei vincoli16

3 Progettazione Logica20

3.1 Introduzione20

3.2 Schema logico20

4 Progettazione Fisica22

4.1 Introduzione22

4.2 Definizione delle tabelle23

4.2.1 Sequence23

4.2.2 Tabella: Laboratorio24

4.2.3 Tabella: Sede24

4.2.4 Tabella: Postazione24

4.2.5 Tabella: Responsabile25

4.2.6 Tabella: Tecnico25

4.2.7 Tabella: Strumento26

4.2.8 Tabella: Prenotazione26

4.2.9 Tabella: Utente26

Capitolo 1

Descrizione del Progetto

* 1. Analisi del problema

L’obiettivo principale del progetto sarà l’implementazione di una base di dati relazionale come supporto per la gestione di prenotazioni di strumenti in sedi di laboratori scientifici.

La base di dati conterrà informazioni su laboratori scientifici e le sedi presenti su tutto il territorio, oltre ad informazioni personali sui dipendenti (tecnici e responsabili) in sedi di laboratorio. Ogni sede avrà un determinato numero di postazioni in cui saranno disponibili strumenti per lo studio in ambito scientifico.

Saranno memorizzate credenziali su utenti registrati, i quali potranno effettuare prenotazioni su strumenti di laboratorio. Le prenotazioni per uno strumento potranno essere effettuate in base a:

* Tipo e descrizione
* Sede di appartenenza

Nel caso in cui ci lo strumento da prenotare in un certo orario sia già associato ad una prenotazione nello stesso giorno con un orario che si sovrappone, allora la prenotazione verrà abortita. Infine, agli utenti sarà possibile gestire le proprie prenotazioni, tramite modifica o cancellazione.

Per ogni strumento sarà visibile un riepilogo, su base mensile ed annuale, di statistiche pertinenti alla frequenza di utilizzo ed agli utenti che lo hanno utilizzato.

Capitolo 2

Progettazione concettuale

2.1 Introduzione

Il seguente capitolo presenta la progettazione del sistema di gestione di laboratori al livello più astratto, che verra rappresentato usando un Class Diagram UML.

Saranno descritte le classi più rilevanti ai requisiti richiesti dal progetto e le associazioni tra esse, oltre ad includere eventuali vincoli.

2.2 Convenzioni nell’uso di UML

Si è scelto di adottare le seguenti convenzioni per rendere la lettura del diagramma UML più leggibile e comprensibile al lettore:

* Tutti gli attributi, ad eccezioni di quelli in cui specificata, hanno molteplicità pari ad [1].
* Le note esterne a classi ed associazioni sono colorate in rosso.
* Tutti i nomi di associazioni sono stati omessi per leggibilità. Saranno definiti all’interno del dizionario di associazioni.

2.3 Class Diagram

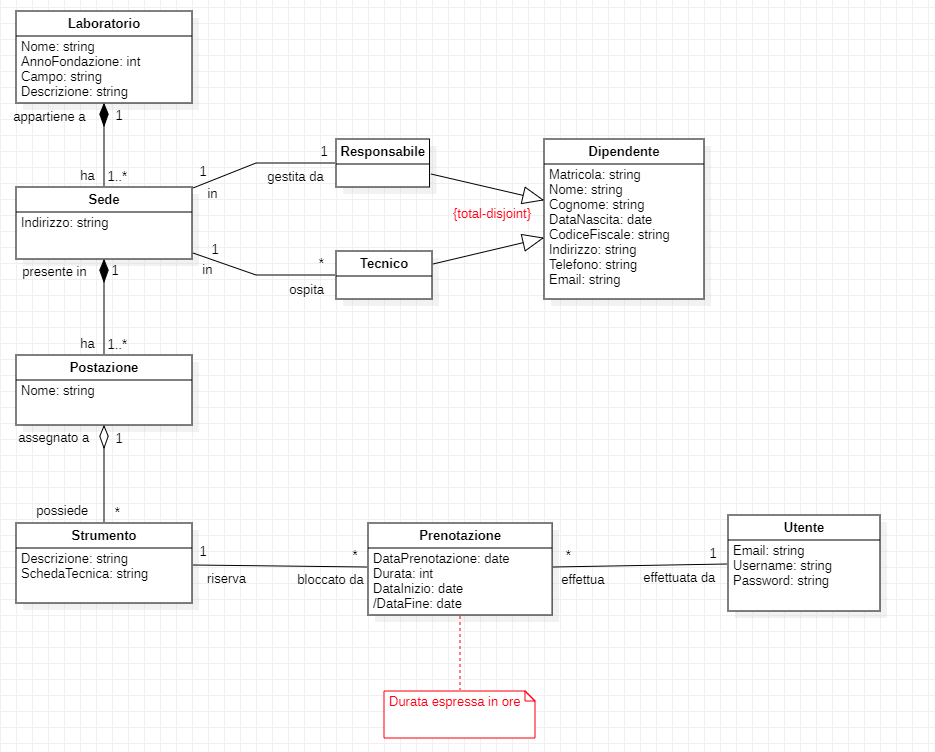


Figura 1: -Class Diagram

2.4 Ristrutturazione

Per tradurre lo schema concettuale in schema relazionale si procederà alla ristrutturazione del Class Diagram precedentemente mostrato. La ristrutturazione sarà necessaria per ricostruire il quadro concettuale in modo da ridurre il livello di astrazione e semplificare il passaggio dalla progettazione concettuale a quella logica e fisica.

Partiremo con l’analisi di ridondanze e di identificativi di classe, seguite dall’eventuale rimozione di attributi strutturati, attributi multipli, composizioni, associazioni e gerarchie di specializzazione.

2.4.1 Analisi delle ridondanze

L’attributo **DataFine** della classe **Prenotazione** è ridondante, siccome è possibile calcolare la data e ora finale della prenotazione tramite gli attributi **DataInizio** e **Durata**. Nei casi in cui sia necessario ottenere la data finale di prenotazione sarà possibile ricavarne il valore aggiungendo la durata(in minuti) alla data iniziale. Pertanto, nel processo di ristrutturazione l’attributo sarà rimosso.

2.4.2 Analisi degli identificativi

Per ogni entità ad eccezione di **Utente** e **Dipendente** verrà aggiunta una chiave surrogata di tipo numerico per identificare univocamente ogni istanza della stessa. Per convenzione avrà la sigla ID\_.

2.4.3 Rimozione degli attributi multipli

L’attributo multiplo **Telefono** di **Dipendente** è un attributo multiplo, dato il caso in cui un dipendente voglia indicare più di un numero telefonico all’interno della sezione dei dati anagrafici. Tuttavia, nella ristrutturazione l’attributo sarà diviso in due attributi singoli: **Telefono1** e **Telefono2**(facoltativo).

2.4.4 Rimozione delle classi di associazione

Non sono presenti classi di associazione.

2.4.5 Rimozione delle gerarchie di specializzazione

La classe generale **Dipendente** sarà compressa sulle due classi specializzate **Responsabile** e **Tecnico**. Le due classi sono strutturalmente simili ma la scelta della divisione di classi porterà ad una gestione più semplice del processo di progettazione fisica.

2.4.6 Rimozione delle composizioni ed aggregazioni

Le composizioni nel Class Diagram sono tra **Laboratorio** e **Sede**, e tra **Sede** e **Postazione.** Saranno sostituite con delle semplici associazioni, tuttavia l’eliminazione di un’istanza della classe totale causerà l’eliminazione di tutte le istanze della classe parte.

L’unica aggregazione è tra le classi **Postazione** e **Strumento**. Uno strumento è contenuto in una postazione ma, non essendo parte fondante di essa, l’eliminazione della postazione non comporterà l’eliminazione degli strumenti associati.

2.5 Class Diagram Ristrutturato

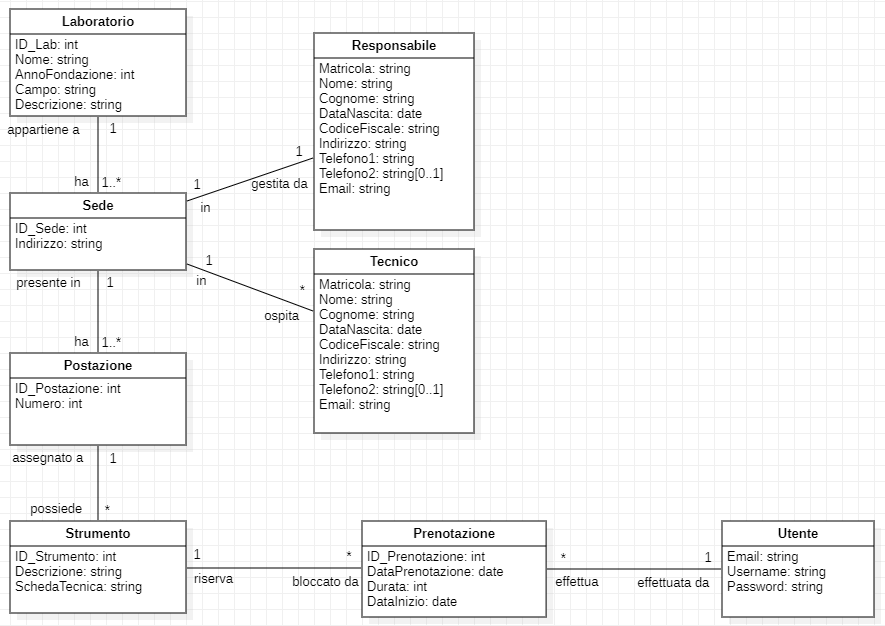


Figura 2: -Class Diagram ristrutturato

2.6 Dizionario delle Classi

|  |  |  |
| --- | --- | --- |
| Classe | Descrizione | Attributi |
| **Laboratorio** | Contiene i dati del laboratorio | **ID\_Lab**(*int*): Identificativo univoco del laboratorio.  **Nome**(*string*):Nome del laboratorio.  **AnnoFondazione**(*int*): Anno in cui è stata fondato il laboratorio.  **Campo**(*string*): Ambito di studi scientifici del laboratorio (es. chimica, biologia).  **Descrizione**(*string*): Descrizione del laboratorio. |
| **Sede** | Contiene i dati sulla sede di un laboratorio | **ID\_Sede**(*int*): Identificativo univoco della sede.  **Indirizzo**(*string*):Indirizzo in cui è situata la sede. |
| **Postazione** | Contiene i dati sulla postazione di una sede | **ID\_Postazione**(*int*): Identificativo univoco della postazione.  **Nome**(*string*): Nominativo associato al luogo fisico della postazione all’interno della sede. |
| **Strumento** | Contiene i dati sullo strumento in una postazione | **ID\_Strumento**(*int*): identificativo univoco dello strumento.  **Descrizione**(*string*): Descrizione dello strumento che ne specifica il nome e le caratteristiche principali.  **SchedaTecnica**(*string*): Scheda tecnica dello strumento. |
| **Responsabile** | Contiene i dati sul responsabile di una sede | **Matricola**(*string*): Numero di matricola del responsabile. Composta dal prefisso “R” seguito da una sequenza di 9 cifre numeriche.  **Nome**(*string*): Nome del responsabile.  **Cognome**(*string*): Cognome del responsabile.  **DataNascita**(*date*): Data di nascita del responsabile.  **CodiceFiscale**(*string*): Codice fiscale del responsabile.  **Indirizzo**(*string*): Indirizzo di residenza del responsabile.  **Telefono1**(*string*): Numero di telefono principale del responsabile.  **Telefono2**(*string*): Numero di telefono secondario del responsabile (opzionale).  **Email**(*string*): Indirizzo email del responsabile. |
| **Tecnico** | Contiene i dati sul tecnico di una sede | **Matricola**(*string*): Numero di matricola del tecnico. Composta dal prefisso “T” seguito da una sequenza di 9 cifre numeriche.  **Nome**(*string*): Nome del tecnico.  **Cognome**(*string*): Cognome del tecnico.  **DataNascita**(*date*): Data di nascita del tecnico.  **CodiceFiscale**(*string*): Codice fiscale del tecnico.  **Indirizzo**(*string*): Indirizzo di residenza del tecnico.  **Telefono1**(*string*): Numero di telefono principale del tecnico.  **Telefono2**(*string*): Numero di telefono secondario del tecnico (opzionale).  **Email**(*string*): Indirizzo email del tecnico. |
| **Prenotazione** | Contiene i dati sulla prenotazione fatta da un utente di uno strumento | **ID\_Prenotazione**(*int*): Identificativo univoco della prenotazione.  **DataPrenotazione**(*date*): Data in cui è stata fatta la prenotazione.  **Durata**(*int*): Durata dell’utilizzo dello strumento prenotato, espressa in ore.  **DataInizio**(*date*): Data in cui si intende utilizzare lo strumento. |
| **Utente** | Contiene i dati di un utente registrato | **Username**(*string*): Nome univoco dell’utente.  **Email**(*string*): Indirizzo email dell’utente.  **Password**(*string*): Password dell’utente. |

2.7 Dizionario delle associazioni

|  |  |  |
| --- | --- | --- |
| Associazione | Descrizione | Classi coinvolte |
| **Appartenenza** | Esprime l’appartenenza di una o più sedi ad un laboratorio | **Laboratorio [1]** ruolo **appartiene a**: indica il laboratorio di cui fa parte la sede.  **Sede [1..\*]** ruolo **ha**: indica le sedi che ha il laboratorio. |
| **Presenza** | Esprime la presenza di una o più postazioni ad una sede | **Sede [1]** ruolo **presente in**: indica la sede di cui fa parte la postazione.  **Postazione [1..\*]** ruolo **ha**: indica le postazioni che ha la sede. |
| **Assegnazione** | Esprime l’assegnazione di uno o più strumenti ad una postazione | **Postazione [1]** ruolo **assegnato a**: indica la postazione a cui è assegnato lo strumento.  **Strumento [1..\*]** ruolo **possiede**: indica gli strumenti che possiede la postazione. |
| **Gestione** | Esprime il responsabile che gestisce la sede | **Responsabile [1]** ruolo **gestita da**: indica il responsabile che gestisce la sede.  **Sede [1]** ruolo **in**: indica la sede che gestisce il responsabile. |
| **Manutenzione** | Esprime l’assegnazione di uno o più tecnici ad una sede | **Tecnico [0..\*]** ruolo **ospita**: indica i tecnici che fanno manutenzione nella sede.  **Sede [1]** ruolo **in**: indica la sede manutenuta dal tecnico. |
| **Riserva** | Esprime le prenotazioni che riservano lo strumento | **Strumento [1]** ruolo **riserva**: indica quale strumento blocca la prenotazione.  **Prenotazione [\*]** ruolo **bloccato da**: indica quale prenotazione blocca lo strumento. |
| **Prenota** | Esprime l’ utente che effettua la prenotazione | **Utente [1]** ruolo **effettuata da**: indica l’utente che fa la prenotazione  **Prenotazione [\*]** ruolo **effettua**: indica le prenotazioni effettuate dall’utente. |

2.8 Dizionario dei vincoli

|  |  |  |
| --- | --- | --- |
| Vincolo | Tipo | Descrizione |
| **compositLab** | Interrelazionale | L’eliminazione di un laboratorio comporterà l’eliminazione di tutte le sedi associate |
| **compositSede** | Interrelazionale | L’eliminazione di una sede comporterà l’eliminazione di tutte le postazioni associate |
| **uniqPostNome** | Intrarelazionale | Il nominativo di postazione sarà unico per la sede |
| **validPostNome** | Dominio | Il nome di postazione dovra’ essere composto da un carattere ‘A-Z’ seguito da una cifra ‘1-9’ |
| **validTel** | Dominio | Non saranno accettati numeri di telefono che non hanno esattamente 10 cifre |
| **validMatr** | Dominio | La matricola dovrà avere esattamente 10 caratteri, con il primo carattere ‘T’(Tecnico) o ‘R’(Responsabile) ed il resto cifre comprese tra ‘0-9’ |
| **validCf** | Dominio | Il codice fiscale dovrà rispettare il formato ufficiale |
| **deleteOrModifyPren** | Interrelazionale | Le prenotazioni potranno essere modificate o cancellate solo se DataInizio è maggiore della data attuale. |
| **validPrenInizio** | Interrelazionale | Le prenotazioni potranno essere effettuate solo se la DataInizio è maggiore della DataPrenotazione |
| **validPrenDurata** | Dominio | Le prenotazioni potranno essere effettuate solo se la Durata è compresa tra 1 ora e 24 ore |
| **noOverlapPren** | Intrarelazionale | Non potranno esistere due prenotazioni per lo stesso strumento con tempi che si sovrappongono |
| **validEmail** | Dominio | Non saranno accettate email che non rispettano la forma: \_@\_.\_ |
| **uniqEmail** | Intrarelazionale | Ogni E-mail sarà univocamente assegnata ad un utente |
| **validUsername** | Dominio | Il nome utente sarà compreso tra 8 e 16 caratteri e potrà contenere solo caratteri contenuti in ‘a-z’, ‘A-Z’e ‘0-9’ |
| **validPw** | Dominio | La password sarà compresa tra 8 e 16 caratteri e dovrà contenere almeno una cifra |

Capitolo 3

Progettazione logica

3.1 Introduzione

In questo capitolo sarà trattata la fase successiva della progettazione della base di dati

scendendo ad un livello di astrazione più basso rispetto alla precedente.

Si tradurrà lo schema concettuale (già predisposto in seguito alla ristrutturazione) in

uno schema logico, dipendente dal tipo di struttura dei dati prescelto cioè quello relazionale puro.

Le PK saranno evidenziate da una singola sottolineatura mentre le FK saranno evidenziate da una doppia sottolineatura

3.2 Schema logico

|  |  |
| --- | --- |
| Laboratorio | (ID\_Lab, Nome, AnnoFondazione, Campo, Descrizione) |
| Sede | (ID\_Sede, Indirizzo, ID\_Lab)  ID\_Lab→Laboratorio.ID\_Lab |
| Postazione | (ID\_Postazione, Nome, ID\_Sede)  ID\_Sede→Sede.ID\_Sede |
| Strumento | (ID\_Strumento, Descrizione, SchedaTecnica, ID\_Postazione)  ID\_Postazione→Postazione.ID\_Postazione |
| Responsabile | (Matricola, Nome, Cognome, DataNascita, CodiceFiscale, Indirizzo, Telefono1, Telefono2, Email, ID\_Sede)  ID\_Sede→Sede.ID\_Sede |
| Tecnico | (Matricola, Nome, Cognome, DataNascita, CodiceFiscale, Indirizzo, Telefono1, Telefono2, Email, ID\_Sede)  ID\_Sede→Sede.ID\_Sede |
| Prenotazione | (ID\_Prenotazione, DataPrenotazione, Durata, DataInizio, ID\_Strumento, Username)  ID\_Strumento→Strumento.ID\_Strumento  Username→Utente.Username |
| Utente | (Email, Username, Password) |

Capitolo 4

Progettazione fisica

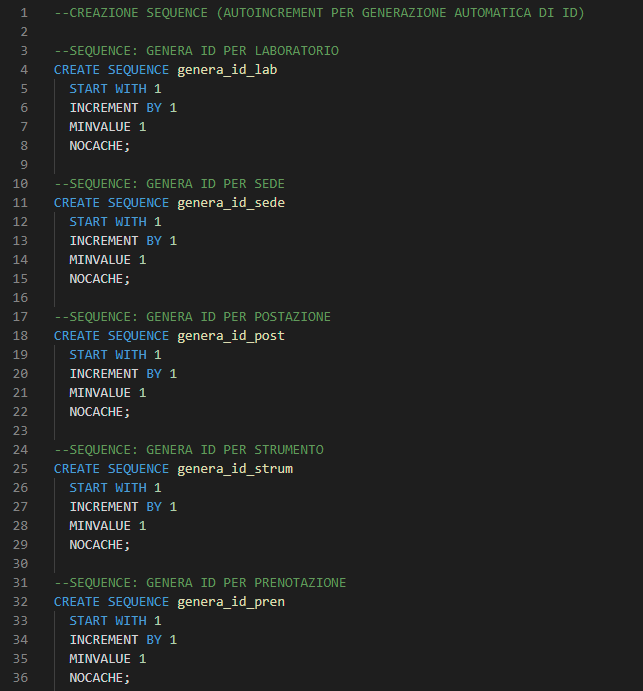
4.1 Introduzione

In questo paragrafo si forniranno le definizioni SQL delle tabelle indicate dallo schema logico, inoltre verranno anche fornite le implementazioni dei vincoli più semplici.

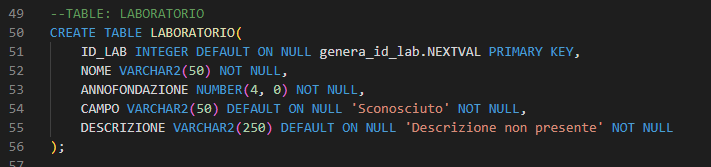
Si osservi che ad ogni tabella prevista di chiave primaria sarà associato un trigger che provvederà a fornire un valore valido a quest’ultima nel caso in cui il suo valore risultasse essere uguale a NULL.

4.2 Definizione delle tabelle

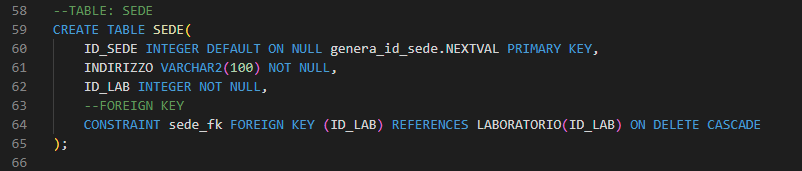
4.2.1 Sequence



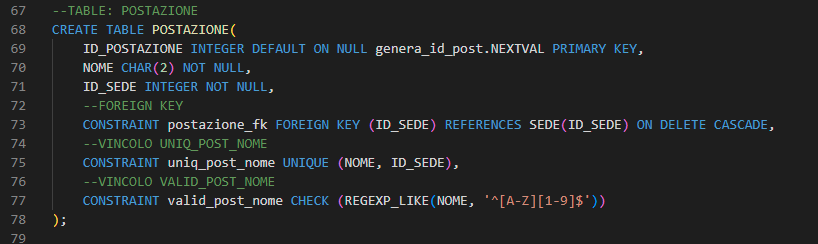
4.2.2 Tabella: Laboratorio



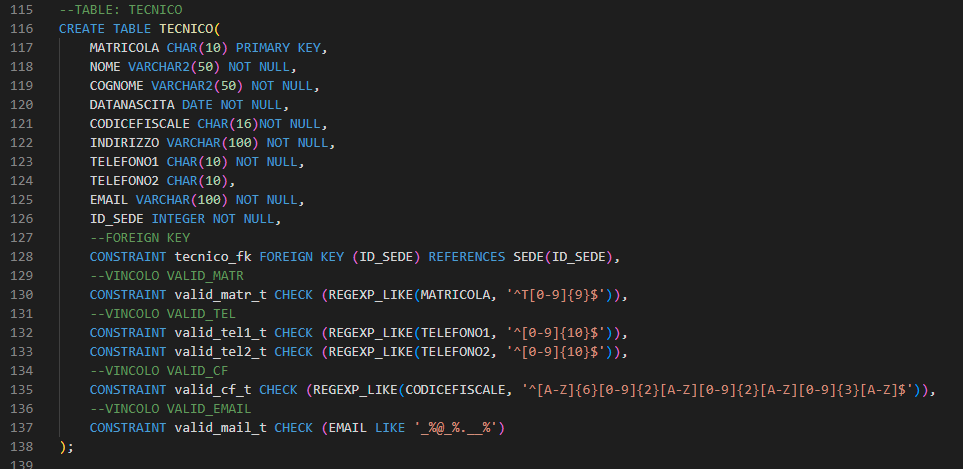
4.2.3 Tabella: Sede



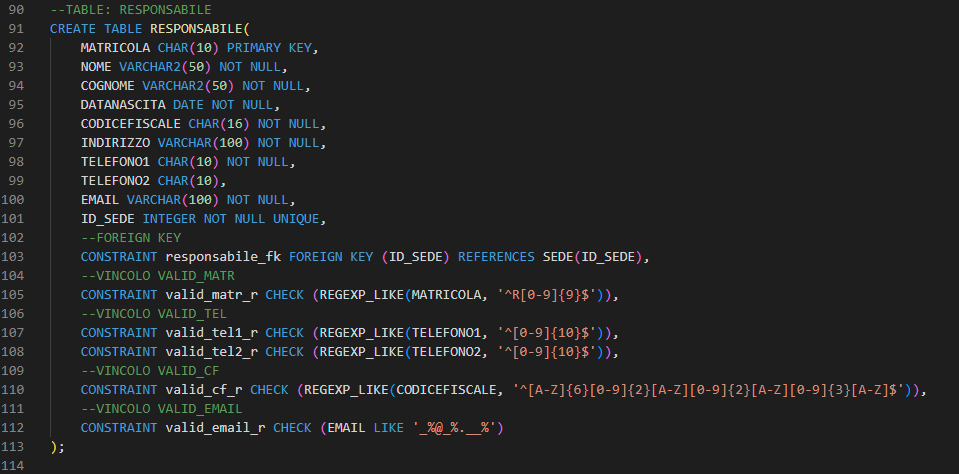
4.2.4 Tabella: Postazione



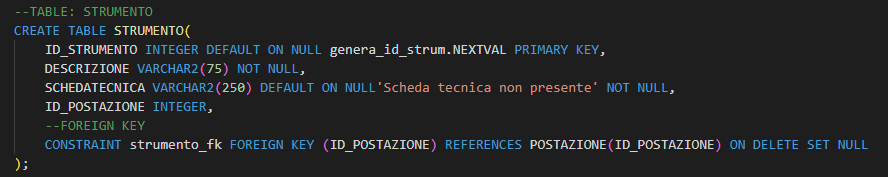
4.2.5 Tabella: Tecnico



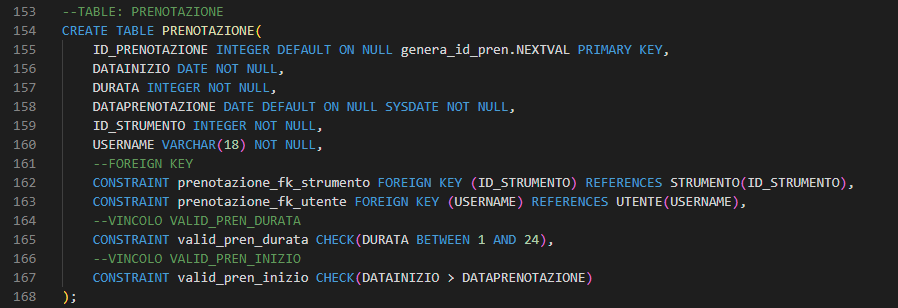
4.2.6 Tabella: Responsabile



4.2.7 Tabella: Strumento



4.2.8 Tabella: Prenotazione



4.2.9 Tabella: Utente

