

Università degli studi di Torino
Basi di Dati
Progetto di laboratorio
A.A. 2020/2021
Bifulco Mario, Capaccetta Giorgio

1. PROGETTAZIONE CONCETTUALE

1.1 Requisiti iniziali

Laboratorio Basi di Dati 2020/21 Progetto di piattaforma vaccinazione COVID

Si vuole realizzare una base di dati per la gestione di una campagna vaccinale su scala nazionale in risposta alla recente epidemia di COVID19 che ha colpito il Paese.

Ogni cittadino è un potenziale vaccinando di cui sono noti nome, cognome, età, città ed indirizzo di residenza, codice fiscale. I cittadini sono ulteriormente suddivisibili secondo l'occupazione in personale sanitario e scolastico, categorie fragili (es, soggetti immunodepressi) e altri.

Sono disponibili tre tipi di vaccino: COVIDIN, CORONAX e FLUSTOP. Ogni vaccino ha un'età minima e massima di somministrazione e una certa efficacia. Ogni vaccino può richiedere una (FLUSTOP) o due (COVIDIN, CORONAX) somministrazioni. Nel caso siano necessarie due somministrazioni, l'intervallo temporale minimo tra prima e seconda dose varia da vaccino a vaccino. Ogni vaccino è organizzato in lotti con data di produzione e scadenza. Di ogni lotto si conoscono le reazioni allergiche eventualmente riscontrate nel Paese.

I centri vaccinali sono distribuiti nelle varie città del Paese a diversi indirizzi e possono esserci più centri per ogni città; ogni centro vaccinale dispone in ogni momento di zero o più fiale di uno o più vaccini ed a ogni centro afferiscono uno o più medici.

Ogni cittadino può fare richiesta di vaccinazione prenotando tramite un sito web o una app per smartphone (Figura 1), lasciando anche un recapito per essere ricontattati. Nel caso del sito web, è richiesto un indirizzo email al posto del numero di cellulare.

The figure displays three sequential screenshots of a mobile application for COVID-19 vaccination registration. Each screen has a status bar at the top showing 'Carrier', signal strength, time '1:20 PM', and battery '100%'. A back arrow and the title 'Prenotazione' are at the top left of each screen. Three numbered circles (1, 2, 3) indicate the current step.

- Screen 1 (Step 1):** Titled 'Compila ed invia i tuoi dati'. It contains three input fields: 'Codice fiscale', 'Categoria', and 'Numero di cellulare'. A 'Prosegui' button is at the bottom. A small note at the bottom reads: 'Creando una prenotazione, aderisci ai termini e condizioni del servizio Italemping'.
- Screen 2 (Step 2):** Titled 'Dichiaro:'. It contains two toggle switches: 'di essere in passato risultato POSITIVO all'infezione da COVID19' (disabled) and 'di presentare allergie a farmaci e/o intolleranze alimentari' (enabled). Below the second toggle is a '+ aggiungi allergia' button. A 'Prosegui' button is at the bottom.
- Screen 3 (Step 3):** Titled 'Controlla i tuoi dati'. It displays the entered information: 'Codice fiscale: RMLRLF80A01L219E', 'Categoria: personale scolastico', and 'Numero di cellulare: 775 5574821'. It also shows the declaration: 'Dichiaro: di essere in passato risultato POSITIVO all'infezione da COVID19' with a 'NO' selection, and 'di presentare allergie a farmaci e/o intolleranze alimentari - amoxicillina' with a 'SI' selection. A 'Prenota' button is at the bottom.

Figura 1

Una task force di operatori riceve le richieste di vaccinazione ed organizza un appuntamento vaccinale come segue.

Ogni cittadino vaccinando può essere indirizzato esclusivamente ad un centro localizzato nella città di residenza. Personale sanitario e personale scolastico possono accedere ai vaccini COVIDIN e CORONAX. I soggetti fragili possono accedere solo a CORONAX e FLUSTOP. Gli altri soggetti non possono accedere a CORONAX.

Nel caso il cittadino abbia avuto precedenti reazioni allergiche, non può essere raccomandato nessun vaccino per cui almeno un lotto abbia registrato una segnalazione di reazione avversa negli ultimi 30 giorni. Il cittadino viene indirizzato al centro più vicino alla residenza dove vi sia disponibilità di almeno una fiala di vaccino compatibile con i vincoli sopra. Nel caso in cui il vaccinando abbia positività pregressa ad un test COVID, viene specificato che la seconda dose del vaccino, se prevista, non deve essere somministrata.

La persona riceve una convocazione al contatto indicato contenente data, ora, centro di vaccinazione, tipo di vaccino da somministrare.

Il cittadino vaccinando, munito di convocazione, si reca in data indicata al centro vaccinale indicato.

Ogni medico è identificato tramite i medesimi dati registrati per i cittadini, e l'afferenza ad un particolare centro vaccinale. I medici si suddividono in medici di base e altri, dove i primi possono somministrare solo vaccini a doppia dose, mentre i secondi sono abilitati anche alla somministrazione di vaccini a singola dose. Al momento dell'arrivo del vaccinando al centro, un medico fra quelli abilitati a somministrare il vaccino richiesto si prende in carico il vaccinando.

In seguito ad ogni vaccinazione, il medico tiene il vaccinato sotto osservazione per 15 minuti per monitorare eventuali effetti allergici relativamente al particolare lotto di vaccino utilizzato. Se non si verificano particolari effetti avversi e il vaccino richiede un richiamo ed il soggetto non ha pregressi di positività COVID, il medico aggiorna la convocazione con un secondo appuntamento per la somministrazione presso la stessa sede dello stesso vaccino in data compatibile con le specifiche del vaccino. Al contrario, se si verifica un effetto allergico, il medico emana un report che indica data e luogo della vaccinazione, tipo vaccino e numero lotto che hanno causato l'allergia al paziente in questione.

Il sistema deve permettere inoltre di eseguire le seguenti operazioni:

- Ogni sera viene stilato un rapporto che indica quante vaccinazioni sono state fatte nella giornata in tutti i centri vaccinali suddivise per categoria di cittadino.
- Ogni sera, ogni centro vaccinale fa l'inventario del numero di dosi ancora disponibili per ogni vaccino.
- Ogni fine settimana, viene stilato un report che indica quante vaccinazioni sono state fatte per ogni vaccino per ognuna delle categorie di cittadini e quante di queste abbiano causato allergie.

1.2 Glossario dei termini

Termini	Descrizione	Sinonimi	Collegamenti
<i>Cittadino</i>	Soggetto generico dell'applicativo	Vaccinando, Persona, Cittadino vaccinando	Personale scolastico e sanitario, Categorie fragili, Medico
<i>Tipo Vaccino</i>	Tutti i tipi di vaccino somministrabili		Vaccino a singola somministrazione, Vaccino a doppia somministrazione
<i>Lotto</i>	Insieme di vaccini organizzati insieme		
<i>Centro vaccinale</i>	Luogo in cui vengono somministrati i vaccini		
<i>Medico</i>	Sottogruppo di cittadini che afferiscono ad un centro vaccinale e sono abilitati alla somministrazione di vaccini		Medico di base, Altro Medico
<i>Altro medico</i>	Sottogruppo dei medici abilitati alla somministrazione di tutti i vaccini		
<i>Medico di base</i>	Sottogruppo dei medici abilitati solo ai vaccini a doppia somministrazione		
<i>Personale scolastico e sanitario</i>	Sottogruppo di cittadini che possono accedere ai vaccini a doppia somministrazione		

<i>Categorie fragili</i>	Sottogruppo di cittadini che possono accedere a un vaccino a doppia somministrazione e a quello a singola somministrazione	Soggetti fragili	
<i>Vaccino a singola dose</i>	Tipologia di vaccino per cui è richiesta una sola somministrazione	FLUSTOP	
<i>Vaccino a doppia dose</i>	Tipologia di vaccino per cui sono necessarie due somministrazioni distanziate da un intervallo di tempo prestabilito	CORONAX, COVIDIN	
<i>Report</i>	È stata riscontrata una reazione allergica e viene segnalata		Vaccino, Lotto, Medico, Centro vaccinale
<i>Appuntamento vaccinale</i>	Informazioni per la somministrazione del vaccino	Convocazione	Cittadino

1.3 Requisiti rivisti

Per **cittadino** rappresentiamo nome, cognome, età, città ed indirizzo di residenza, codice fiscale, numero di cellulare o indirizzo email, allergie, precedente positività.

I cittadini sono divisibili in tipologie: personale sanitario, personale scolastico, categorie fragili, altri. Un cittadino del tipo “personale sanitario” può essere un medico.

Per **medico** rappresentiamo il centro vaccinale di appartenenza.

I medici sono suddivisi in due tipologie, medici di base e altri medici. Un medico di base può somministrare solo vaccini di tipologia “doppia dose”. Gli altri medici possono somministrare tutti i vaccini.

Per un **tipo vaccino** rappresentiamo età minima, età massima e efficacia.

Sono disponibili 3 vaccini: COVIDIN, CORONAX, FLUSTOP. I vaccini sono divisibili in tipologie: singola somministrazione (flustop), doppia somministrazione (covidin e coronax). Per i vaccini a doppia somministrazione rappresentiamo l’intervallo tra la prima e la seconda somministrazione. I vaccini sono contenuti in lotti.

Per **lotto** rappresentiamo data di produzione, data di scadenza e reazioni allergiche riscontrate.

Se viene riscontrata una reazione allergica per un vaccino di un lotto nessun vaccino di quel lotto può essere somministrato a cittadini con quell’allergia per i successivi 30 giorni

Per **centro vaccinale** rappresentiamo città, indirizzo, dosi disponibili per ogni vaccino, medici di appartenenza.

Per **appuntamento vaccinale** rappresentiamo il cittadino di riferimento, il vaccino che verrà somministrato in un centro vaccinale della propria città, data e ora della somministrazione.

Ai cittadini di tipo “personale sanitario” e “personale scolastico” vengono somministrati vaccini di nome “covidin” e “coronax”, ai cittadini di tipo “soggetti fragili” vengono somministrati vaccini di nome “coronax” e “flustop”, agli altri cittadini vengono somministrati vaccini di nome “covidin” e “flustop”. Se il vaccino somministrato è di tipo “doppia somministrazione” e il cittadino non ha avuto precedente positività al COVID e non è stata riscontrata una reazione allergica viene organizzato un secondo appuntamento vaccinale nella stessa sede e in data compatibile con l’intervallo di somministrazione del vaccino.

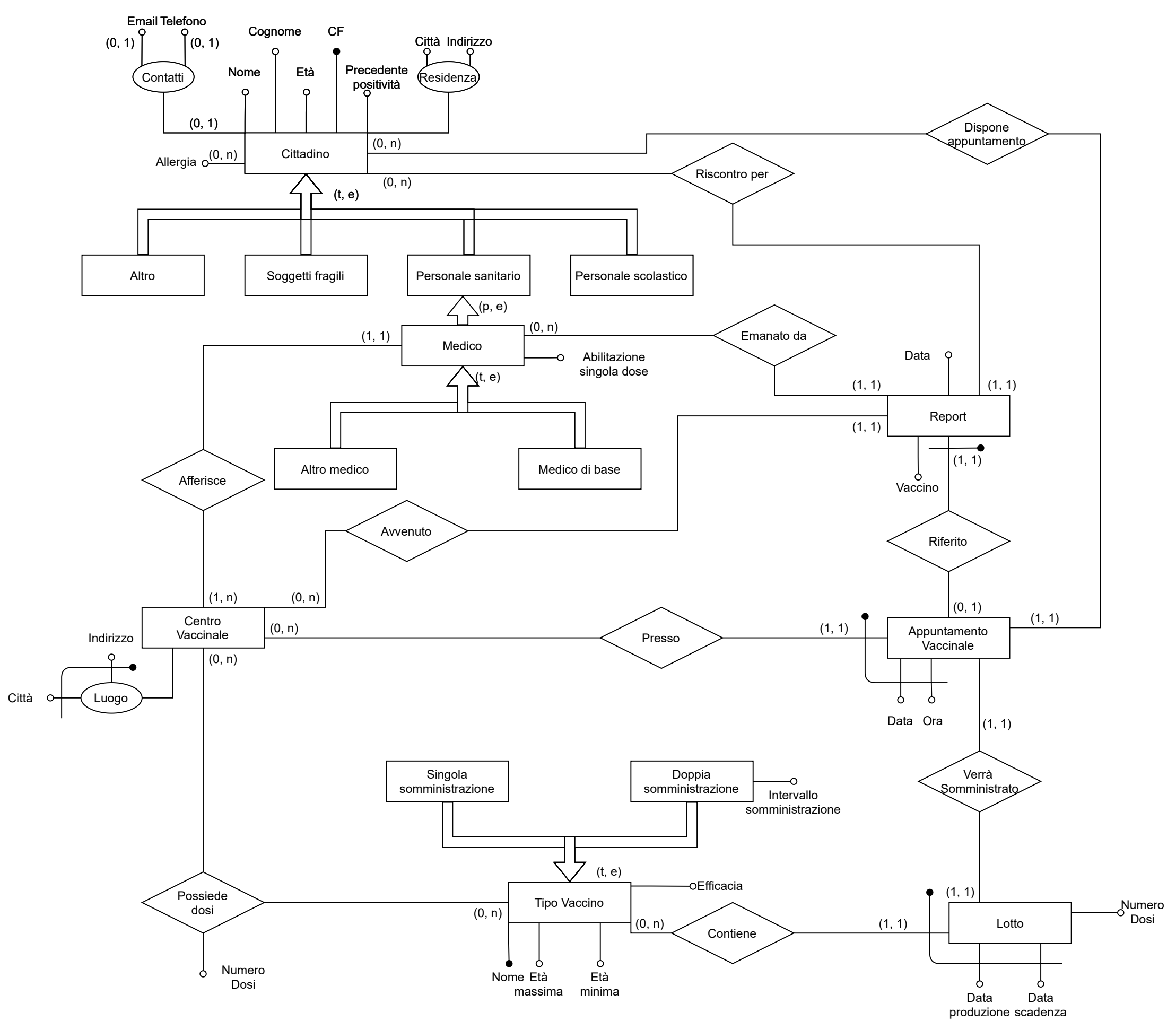
Per **report** rappresentiamo data, luogo, nome del vaccino, numero del lotto che ha causato la reazione allergica e cittadino che ha riscontrato l’allergia

Il report viene emanato da un medico in caso di riscontro allergico.

1.4 Schema E-R

Business rules:

- Un cittadino può avere appuntamenti vaccinali solo per centri della sua città che possiedono dosi del tipo di vaccino a cui può accedere
- I soggetti fragili possono ricevere vaccini con nome coronax e flustop, il personale sanitario e scolastico può ricevere vaccini con nome covidin e coronax, gli altri possono ricevere vaccini con nome covidin e flustop
- Abilitazione singola dose è true solo se è un medico è in realtà un'istanza di altro medico
- Email e Telefono sono valorizzati in mutua esclusione



2 PROGETTAZIONE LOGICA

2.1 Tavola dei volumi

Concetto	Tipo	Volume	Note
<i>Tipo Vaccino</i>	E	3	
<i>Singola somministrazione</i>	E	1	
<i>Doppia somministrazione</i>	E	2	
<i>Lotto</i>	E	200 000	Considerando una media di 100 fiale a lotto e 5 dosi a fiala
<i>Report</i>	E	1 100	Considerando una media di 11 casi di reazioni allergiche ogni milione di somministrazioni
<i>Cittadino</i>	E	60 000 000	
<i>Personale scolastico</i>	E	850 000	
<i>Personale sanitario</i>	E	725 000	
<i>Medico</i>	E	450 000	
<i>Altro medico</i>	E	400 000	
<i>Medico di base</i>	E	50 000	
<i>Soggetti fragili</i>	E	14 000 000	
<i>Altri</i>	E	60 000 000	
<i>Appuntamento vaccinale</i>	E	100 000 000	Considerando un appuntamento vaccinale per ogni dose
<i>Centro vaccinale</i>	E	3 000	Considerando i dati forniti dal ministero della salute
<i>Possiede Dosi</i>	A	500 000	
<i>Contiene</i>	A	200 000	
<i>Riferito</i>	A	1 100	

<i>Verrà somministrato</i>	A	100 000 000	
<i>Dispone appuntamento</i>	A	100 000 000	
<i>Afferisce</i>	A	450 000	
<i>Presso</i>	A	100 000 000	
<i>Avvenuto</i>	A	1 100	
<i>Emanato da</i>	A	1 100	
<i>Riscontro per</i>	A	1 100	

2.2 Tavola delle operazioni

Operazione	Tipo	Frequenza
Quantità di vaccinazioni effettuate per categoria	B	1/giorno
Inventario delle dosi disponibili per ogni vaccino in ogni centro	B	1/giorno
Vaccinazioni fatte per ogni vaccino divise per categoria dei cittadini e quante reazioni allergiche sono state riscontrate	B	1/settimana
Prenotazione effettuata	I	500000/giorno
Riduzioni dosi disponibili in un centro per un dato tipo di vaccino	I	500000/giorno
Creazione di un report per riscontro allergico	I	1/settimana
Registrazione utente	I	300000/giorno

2.3 Ristrutturazione dello schema E-R

2.3.1 Analisi delle ridondanze

Vaccino in REPORT è superfluo e ricavabile tramite tre passaggi. RIFERITO che collega a APPUNTAMENTO VACCINALE, VERRÀ SOMMINISTRATO che collega a LOTTO e CONTIENE che collega a TIPO VACCINO.

Considerando $CARD(REPORT) = 1\ 100$

Senza la ridondanza servono 4 letture, da REPORT a APPUNTAMENTO VACCINALE, da APPUNTAMENTO VACCINALE a LOTTO, da LOTTO a TIPO VACCINO e da TIPO VACCINO ricaviamo la tipologia.

Tempo: $4 * 1\ 100 = 4\ 400$

Spazio: 0 Byte

Con la ridondanza per ottenere il tipo di vaccino basta una lettura perché l'attributo è già in report.

Tempo: $1 * 1\ 100 = 1\ 100$ letture a settimana

Spazio: immaginando di salvare il tipo del vaccino non come stringa ma come numero intero (e quindi enumerare i valori possibili) basterebbero 2 bit per rappresentare i valori possibili, anche prevedendone 3 per lasciare la possibilità di aggiungere tipi di vaccino in futuro

$3\ \text{bit} * 1\ 100\ \text{entità} = 3\ 300\ \text{bit} = \text{circa } 410\ \text{byte}$

A fronte dell'irrisorio spazio aggiuntivo richiesto per mantenere la ridondanza risulta conveniente conservarla per ridurre gli accessi giornalieri di letture al database.

REPORT presenta altre due ridondanze, RISCONTRO PER e AVVENUTO, entrambe deducibili passando per APPUNTAMENTO VACCINALE tramite RIFERITO e poi con le associazioni PRESSO per ottenere il centro vaccinale di riferimento e DISPONE APPUNTAMENTO per verificare il cittadino.

Senza ridondanza accorrono $2 * 1\ 100$ letture (2 200), per reperire i centri di riferimento dei report, mentre con la ridondanza le letture si riducono a 1 100.

Considerando $CARD(CENTRO\ VACCINALE) = 3\ 000$ per rappresentare la ridondanza servono almeno 12 bit, quindi: $12\ \text{bit} * 3\ 000\ \text{entità} = 36\ 000\ \text{bit} = 4\ 500\ \text{byte} = \text{circa } 4.4\ \text{Kbyte}$

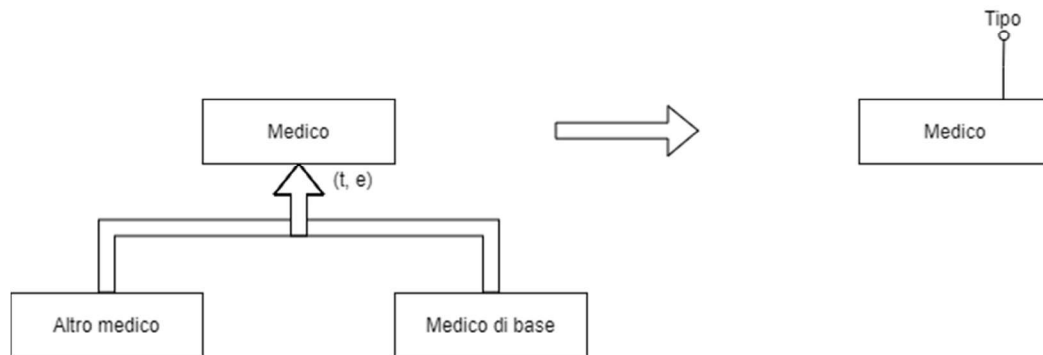
Poco meno di 4.5 Kbyte sono un quantitativo irrisorio di dati, quindi ha senso mantenere la ridondanza.

Considerando $CARD(CITTADINO) = 60\ 000\ 000$ per rappresentare la ridondanza servono almeno 26 bit, quindi $26\ \text{bit} * 60\ 000\ 000\ \text{entità} = 1\ 560\ 000\ 000\ \text{bit} = 195\ 000\ 000\ \text{Byte} = \text{circa } 190\ 429.7\ \text{Kbyte} = 185.97\ \text{Mbyte}$

Rispetto alle ridondanze precedenti questa ha una mole maggiore di dati da mantenere coerente, visto il basso numero di letture si può optare per rimuovere la ridondanza.

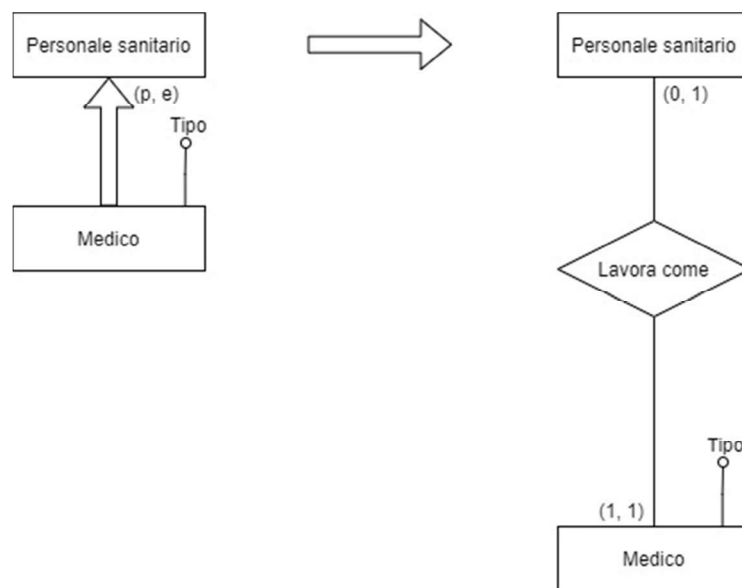
2.3.2 Eliminazione delle generalizzazioni

Eliminazione della generalizzazione di Medico:



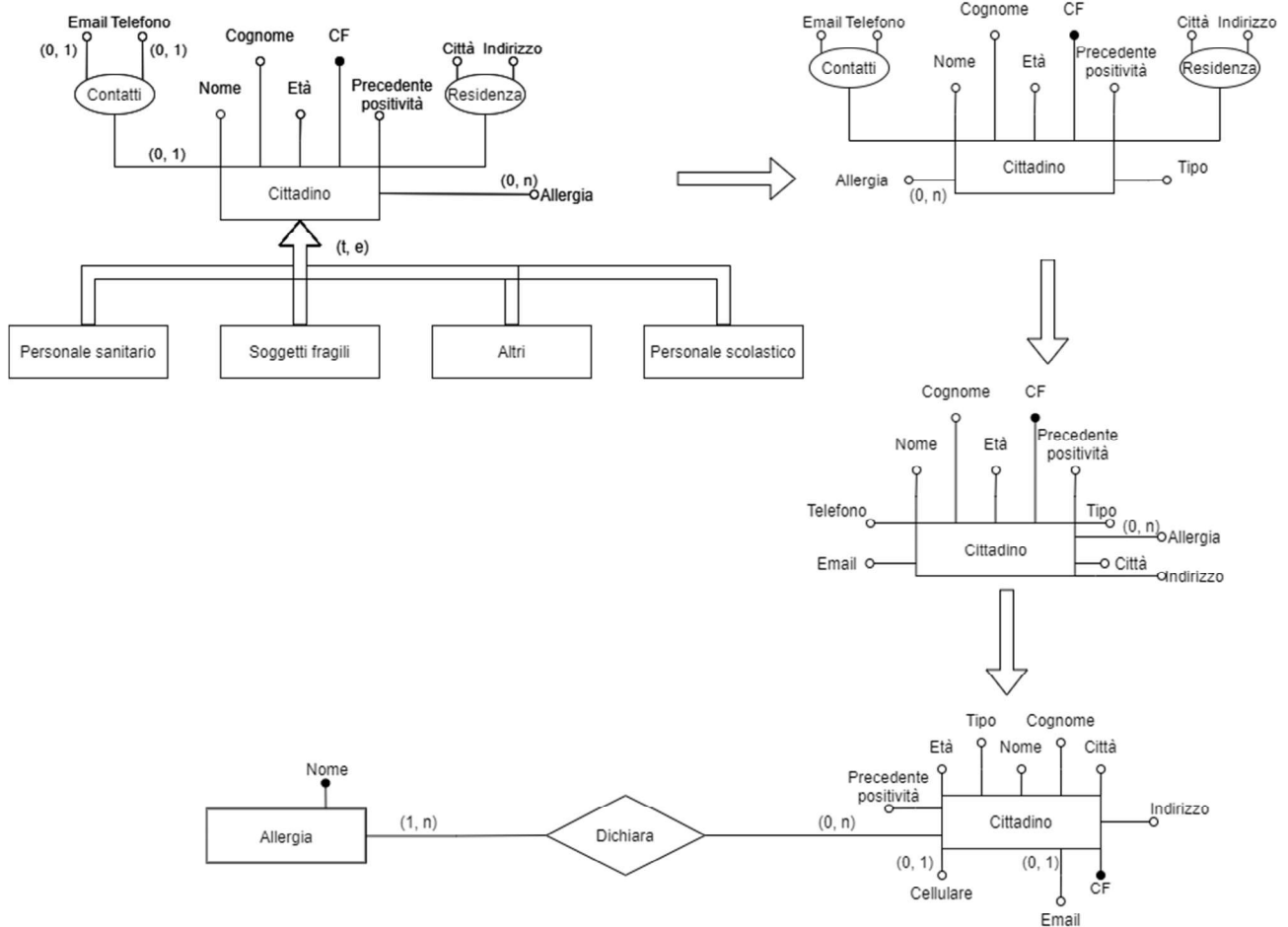
Tutte le informazioni dei figli sono esprimibili tramite un attributo su Medico, avendo poi cura di controllare la coerenza delle operazioni legali.

Eliminazione della generalizzazione Personale sanitario



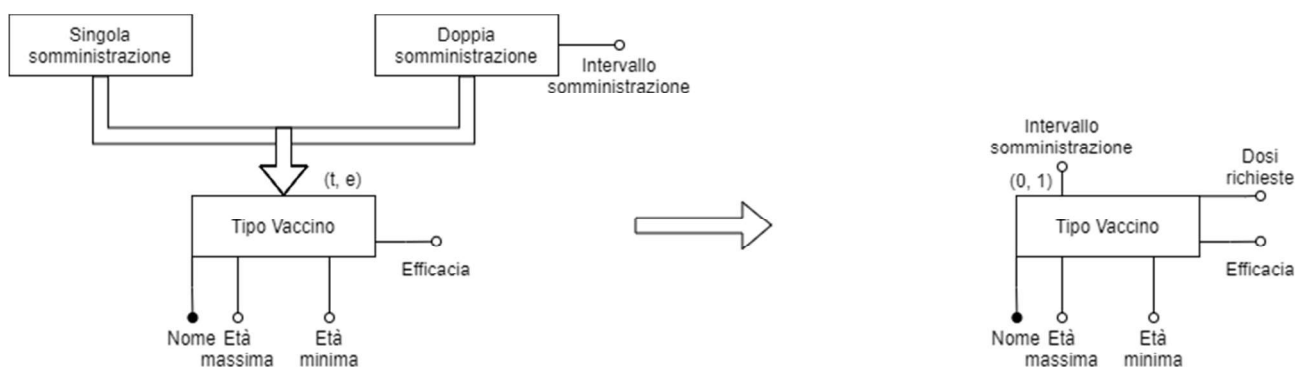
Al posto di una generalizzazione è preferibile un'associazione che costituisce un controllo sulle operazioni del cittadino specializzato.

Eliminazione della generalizzazione di Cittadino



I figli di Cittadino sono accorpabili al genitore aggiungendo un attributo che mantiene l'informazione sulla categoria di cittadino.
L'attributo multi valore viene reificato in entità.

Eliminazione della generalizzazione di Vaccino



Come per Cittadino i figli non aggiungono informazioni essenziali che non possano essere espresse nel padre aggiungendo attributi.

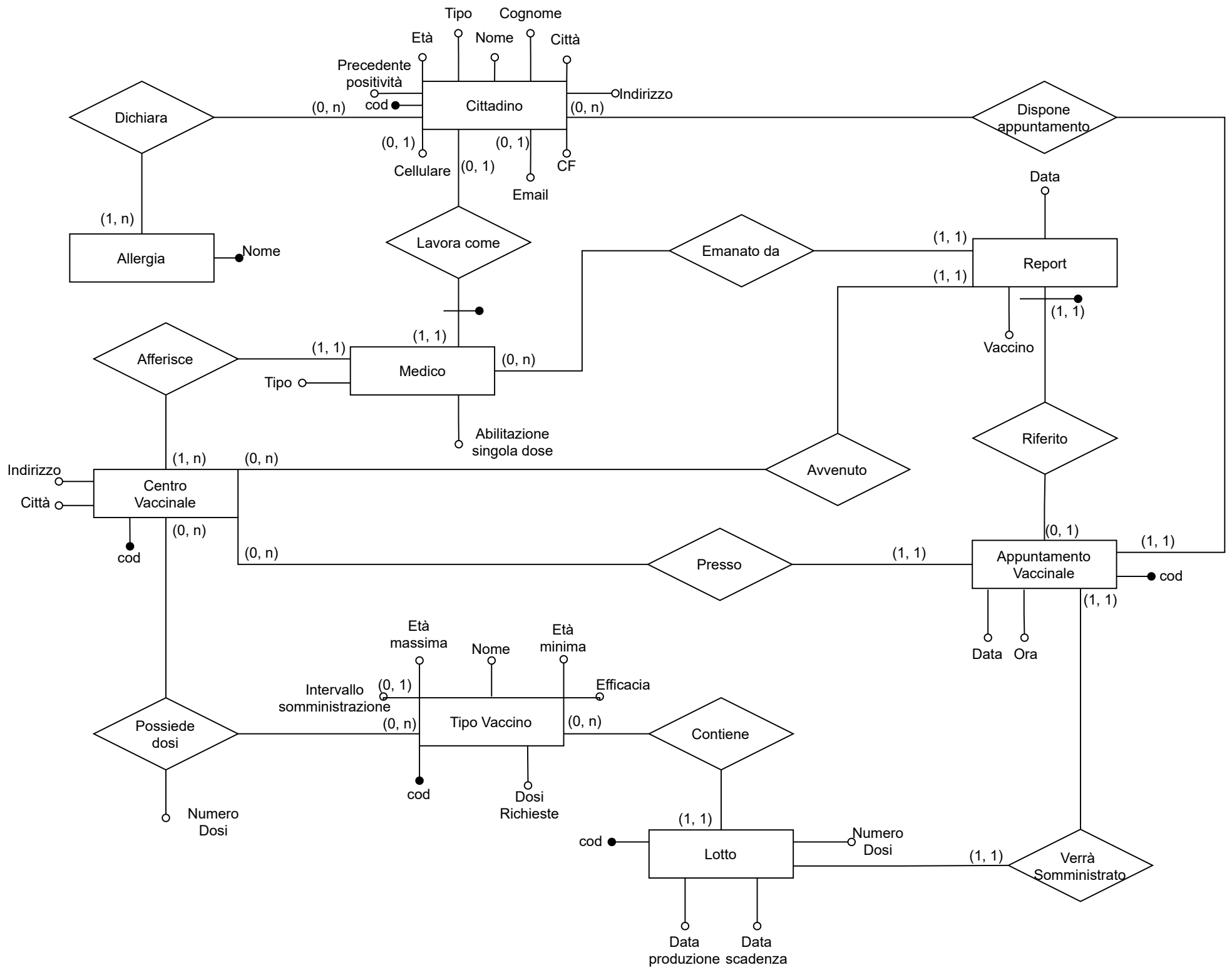
2.3.4 Scelta degli identificatori principali

Entità	Identificativo	Motivazione
<i>Tipo Vaccino</i>	Cod	La tipologia dei vaccini viene utilizzata in molte relazioni della base di dati, risulta quindi più efficiente riferirsi ad essa tramite un identificativo numerico
<i>Lotto</i>	Cod	Identificativo aggiunto manualmente per evitare una chiave che comprendesse quasi tutti gli attributi dell'entità
<i>Report</i>	Riferito	L'associazione Riferito con Appuntamento Vaccinale permette di identificare in modo univoco ogni report, visto che i report non sono richiamati in altre tabelle non è necessario l'uso di identificativi sintetici
<i>Cittadino</i>	Cod	Il codice fiscale identifica in modo univoco un cittadino Italiano, tuttavia risulta dispendioso da gestire in quanto viene memorizzato come stringa, l'uso di un identificativo numerico risulta migliore sia in termini di spazio che in termini di tempo
<i>Medico</i>	Cod	Il codice di medico è un riferimento al codice di cittadino
<i>Appuntamento vaccinale</i>	Cod	La chiave Data, Ora, associazione Presso risulta scomoda da usare per riferirsi ad appuntamento vaccinale, in quanto comprende buona parte delle informazioni dell'entità, per facilitarne l'interazione e per leggerezza nella base di dati si inserisce un identificativo sintetico
<i>Centro vaccinale</i>	Cod	Per non appesantire troppo le relazioni che fanno riferimento a centro vaccinale è opportuno introdurre un identificativo sintetico
<i>Allergia</i>	Nome	Il nome è univoco e non serve inserire id per ragioni di efficienza

2.4 Schema E-R ristrutturato

Business rules:

- Un cittadino può avere appuntamenti vaccinali solo per centri della sua città che possiedono dosi del tipo di vaccino a cui può accedere
- I cittadini con tipo "soggetti fragili" possono ricevere solo vaccino con nome coronax e flustop, i cittadini con tipo "personale sanitario" e "personale scolastico" possono ricevere solo vaccini con nome covidin e coronax, gli altri possono ricevere solo vaccini con nome covidin e flustop.
- Abilitazione singola dose è true solo se il tipo di medico è "altro medico"
- Intervallo somministrazione può essere valorizzato solo se le dosi richieste sono più di una, negli altri casi è nullo
- E-mail e Telefono sono valorizzati in mutua esclusione
- L'associazione "Lavora come" tra Medico e Cittadino è possibile solo se il Cittadino è di tipo "personale sanitario"



2.5 Schema relazionale

CITTADINO(Cod, CF, Nome, Cognome, Et , Email*, Cellulare*, Indirizzo, Citt , PrecedentePositivit , Tipo)

CENTROVACCINALE(Cod, Citt , Indirizzo)

MEDICO(Cod, Tipo, CentroVaccinale, AbilitazioneSingolaDose)

LOTTO(Cod, TipoVaccino, DataScadenza, DataProduzione, NumeroDosi)

TIPOVACCINO(Cod, Nome, Et Min, Et Max, Efficacia, DosiRichieste, IntervalloSomministrazione*)

APPUNTAMENTO VACCINALE(Cod, Data, Ora, CentroVaccinale, Cittadino, Lotto)

POSSIEDE DOSI(CentroVaccinale, TipoVaccino, NumeroDosi)

ALLERGIA(Nome, Cittadino)

REPORT(AppuntamentoVaccinale, Medico, CentroVaccinale, Vaccino, Data)

Vincoli di integrit  referenziale:

ALLERGIA(Cittadino) referencia CITTADINO(Cod)

REPORT(AppuntamentoVaccinale) referencia APPUNTAMENTO VACCINALE(Cod)

REPORT(Medico) referencia MEDICO(Cod)

REPORT(CentroVaccinale) referencia CENTRO VACCINALE(Cod)

REPORT(Vaccino) referencia TIPO VACCINO(Cod)

APPUNTAMENTO VACCINALE(Cittadino) referencia CITTADINO(Cod)

APPUNTAMENTO VACCINALE(CentroVaccinale) referencia CENTRO VACCINALE(Cod)

APPUNTAMENTO VACCINALE(Lotto) referencia LOTTO(Cod)

MEDICO(CentroVaccinale) referencia CENTRO VACCINALE(Cod)

MEDICO(Cod) referencia CITTADINO(Cod)

POSSIEDE DOSI(CentroVaccinale) referencia CENTRO VACCINALE(Cod)

POSSIEDE DOSI(TipoVaccino) referencia TIPOVACCINO(Cod)

LOTTO(TipoVaccino) referencia TIPO VACCINO(Cod)

3 IMPLEMENTAZIONE

3.1 DDL di creazione database

-- Tipi

```
create type tipo_cittadino as ENUM('personale sanitario', 'personale scolastico',  
  'soggetto fragile', 'altro');  
create type tipo_medico as ENUM('altro medico', 'medico di base');  
create type nome_vaccino as ENUM('Covidin', 'Coronax', 'Flustop');
```

-- Sequenze

```
create sequence if not exists id_vaccino  
increment by 1 start 1;  
create sequence if not exists id_cittadino  
increment by 1 start 1;  
create sequence if not exists id_appuntamento_vaccinale  
increment by 1 start 1;  
create sequence if not exists id_centro  
increment by 1 start 1;
```

-- Tabelle

```
create table if not exists cittadino (  
  cod integer default nextval('id_cittadino') primary key,  
  cf varchar(16) not null check (length(cf)=16) unique,  
  nome varchar(32) not null,  
  cognome varchar(32) not null,  
  eta integer not null check (eta>=0),  
  email varchar(128) unique,  
  cellulare bigint unique check (cellulare>3200000000 and cellulare<3939999999),  
  indirizzo varchar(128) not null,  
  citta varchar(64) not null,  
  precedente_positivita boolean not null default FALSE,  
  tipo tipo_cittadino not null default 'altro',  
  check (cellulare is not null or email is not null)  
);
```

/* Nota sugli attributi città: Al fine di evitare la ripetizione di una stringa per decine di migliaia di volte, in un contesto reale sarebbe da valutare la creazione di una tabella città (con id, nome e opzionalmente altri parametri) a cui riferirsi tramite id

Discorsi analogo potrebbe essere fatto con l'attributo indirizzo, creando un'entità indirizzo collegata a città tramite associazione tuttavia in questo caso il risparmio in termini di spazio sarebbe minore in quanto VAL(indirizzo, cittadino) << VAL(città, cittadino)

Il discorso è analogo in tutti i casi in cui trattiamo le città o gli indirizzi

*/

```
create table if not exists centro_vaccinale (  
  cod integer default nextval('id_centro') primary key,  
  indirizzo varchar(128) not null,  
  citta varchar(64) not null,  
  unique (indirizzo, citta)  
);
```

```
create table if not exists tipo_vaccino (  
  cod integer default nextval('id_vaccino') primary key,  
  nome varchar(32) unique,  
  eta_min integer not null check (eta_min>=0),  
  eta_max integer not null check (eta_max>eta_min),  
  efficacia real not null check (efficacia>0 and efficacia<=100),  
  dosi_richieste integer not null check (dosi_richieste=1 or dosi_richieste=2),  
  intervallo_somministrazione integer default null check (intervallo_somministrazione>0  
    or intervallo_somministrazione is null),  
  check (  
    case  
      when dosi_richieste=1 then intervallo_somministrazione is null  
      else intervallo_somministrazione is not null  
    end  
  )  
);
```

```
create table if not exists medico (  
  cod integer primary key,  
  tipo tipo_medico not null,  
  centro integer not null,  
  abilitazione_singola_dose boolean not null,  
  foreign key (cod) references cittadino (cod) on delete cascade on update cascade,  
  foreign key (centro) references centro_vaccinale(cod) on update cascade,  
  check (  
    case  
      when abilitazione_singola_dose=TRUE then tipo='altro medico'  
      else tipo='medico di base'  
    end  
  )  
);
```



```

create table if not exists lotto (
  cod varchar(6) check (length(cod)=6) primary key,
  -- supponiamo che gli id siano alfanumerici e di lunghezza costante
  vaccino integer,
  num_dosi integer not null default 500 check (num_dosi>0),
  data_produzione date not null,
  data_scadenza date not null check (data_scadenza > data_produzione),
  unique (data_produzione, data_scadenza, vaccino),
  foreign key (vaccino) references tipo_vaccino (cod) on update cascade
);

```

```

create table if not exists appuntamento_vaccinale (
  cod integer default nextval('id_appuntamento_vaccinale') primary key,
  data_appuntamento date,
  ora time,
  centro integer,
  lotto varchar(6) not null,
  cittadino integer not null,
  /*
  si ipotizza che gli appuntamenti vaccinali siano creati in funzione del cittadino
  e non che vengano prima creati i vari appuntamenti e in un secondo momento associati ai cittadini
  */
  foreign key (centro) references centro_vaccinale (cod) on update cascade,
  foreign key (lotto) references lotto(cod) on update cascade,
  foreign key (cittadino) references cittadino (cod) on delete cascade on update cascade,
  unique (data_appuntamento, ora, centro)
);

```

```

create table if not exists possiede_dosi (
  centro integer,
  vaccino integer,
  num_dosi integer not null check (num_dosi>=0),
  -- da rendere consistente, ogni volta che viene somministrata una dose va sottratto 1
  primary key (centro, vaccino),
  foreign key (centro) references centro_vaccinale (cod) on update cascade,
  foreign key (vaccino) references tipo_vaccino (cod) on update cascade
);

```

```

create table if not exists allergia (
  nome varchar(128) not null,
  cittadino integer,
  foreign key (cittadino) references cittadino (cod) on delete cascade on update cascade
);

```

```
create table if not exists report (  
  appuntamento_vaccinale integer not null,  
  medico integer not null,  
  centro integer not null,  
  data_report date not null,  
  vaccino integer not null,  
  foreign key (centro) references centro_vaccinale (cod) on update cascade,  
  foreign key (vaccino) references tipo_vaccino (cod) on update cascade,  
  foreign key (medico) references medico (cod) on update cascade,  
  foreign key (appuntamento_vaccinale) references appuntamento_vaccinale (cod),  
  primary key (appuntamento_vaccinale)  
);
```

3.2 DML di popolamento delle tabelle

begin;

-- Cittadino

-- nota: essendo dati fittizi si sceglie di mettere come
suffisso di CF a11a123b (o 51 nel caso di donne)

-- e rendere coerente solo la prima parte

insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)

values ('dptfri11a11a123b', 'Fiero', 'Di Pietro', 10, null, 3290145027, 'Via del sole 7',
'Torino', FALSE, 'altro');

insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)

values ('brmpal01a11a123b', 'Palamede', 'Bramati', 20, 'p.bramati@email.com', null,
'Strada portorosso 12', 'Genova', TRUE, 'personale sanitario');

insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)

values ('ghmngl98a51a123b', 'Angela', 'Aghemio', 23, null, 3376295481,
'Corso san giovanni 138', 'Roma', FALSE, 'personale scolastico');

insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)

values ('smrlni60a11a123b', 'Lino', 'Sommariva', 61, null, 3292674805, 'Stradale torino',
'Milano', FALSE, 'soggetto fragile');

insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)

values ('tnorsl76a51a123b', 'Rossella', 'Tonani', 45, 'ross.tonani@gmail.com', null,
'Via tempesta 5', 'Milano', TRUE, 'altro');

insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)

values ('zpagbr91a11a123b', 'Gabriele', 'Zappa', 30, null, 3902837401, 'Via alighieri 1a',
'Napoli', FALSE, 'altro');

insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)

values ('dflflp95a11a123b', 'Filippo', 'De Filippi', 26, 'defilippi.fil@yahoo.com', null,
'Via de medici 90', 'Napoli', TRUE, 'altro');

insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)

values ('zmllei70a11a123b', 'Elio', 'Zemello', 51, 'zemee51@libero.it', null, 'Via arsenale 5',
'Milano', FALSE, 'personale sanitario');

insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)

values ('stfbrd86a11a123b', 'Berardo', 'Stefanelli', 35, null, 3290394850, 'Via garibaldi 10',

```

'Torino', TRUE, 'personale scolastico');
insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)
values ('sprsmn90a11a123b', 'Simone', 'Sparaventi', 31, 'ventisimone_31@email.net', null,
'Via sapienza 50', 'Roma', FALSE, 'altro');
insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)
values ('vsclrn40a11a123b', 'Lorenzo', 'Viscomi', 81, null, 3592531876, 'Via del sentiero',
'Milano', TRUE, 'soggetto fragile');
insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)
values ('mzania69a11a123b', 'Iana', 'Mazzuia', 52, 'maziana_2@gmail.com', null,
'Strada della repubblica', 'Torino', FALSE, 'altro');
insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)
values ('brcpio85a11a123b', 'Pippo', 'Burchi', 36, null, 3236987459, 'Corso caselle 70',
'Torino', TRUE, 'personale sanitario');
insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)
values ('flnlmr66a11a123b', 'Leonardo', 'Filano', 55, 'fil.leo66@gmail.co,', null,
'Corso cesare 177', 'Milano', FALSE, 'personale sanitario');
insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)
values ('dgzrti85a51a123b', 'Rita', 'De Guzzi', 36, null, 3596542587, 'Via dei mercanti 8',
'Genova', FALSE, 'altro');
insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)
values ('mntfda60a51a123b', 'Fadia', 'Menta', 61,
'f.menta@virgilio.it', null, 'Strada del colle 3', 'Roma', TRUE, 'personale sanitario');
insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)
values ('dgvant70a11a123b', 'Antonio', 'De Giovanni', 51, null, 3549631478,
'Corso romano 75', 'Napoli', FALSE, 'personale sanitario');
insert into cittadino (cf, nome, cognome, eta, email, cellulare, indirizzo, citta,
precedente_positivita, tipo)
values ('ctsmta89a11a123b', 'Matteo', 'Actis', 32,
'mattewactis@gmail.com', null, 'Via del mare 5', 'Genova', TRUE, 'personale sanitario');

```

-- Centro Vaccinale

```
insert into centro_vaccinale (indirizzo, citta)
values ('Via ritrovata 15', 'Torino');
insert into centro_vaccinale (indirizzo, citta)
values ('Via retta 87', 'Milano');
insert into centro_vaccinale (indirizzo, citta)
values ('Via Roma 1', 'Roma');
insert into centro_vaccinale (indirizzo, citta)
values ('Via della speranza 42', 'Napoli');
insert into centro_vaccinale (indirizzo, citta)
values ('Via degli artisti 7', 'Genova');
```

-- Medico

```
insert into medico (cod, tipo, centro, abilitazione_singola_dose)
values (8, 'medico di base', 2, FALSE);
insert into medico (cod, tipo, centro, abilitazione_singola_dose)
values (14, 'altro medico', 2, TRUE);
insert into medico (cod, tipo, centro, abilitazione_singola_dose)
values (13, 'medico di base', 1, FALSE);
insert into medico (cod, tipo, centro, abilitazione_singola_dose)
values (16, 'altro medico', 3, TRUE);
insert into medico (cod, tipo, centro, abilitazione_singola_dose)
values (17, 'altro medico', 4, TRUE);
insert into medico (cod, tipo, centro, abilitazione_singola_dose)
values (18, 'altro medico', 5, TRUE);
```

-- Vaccino

```
insert into tipo_vaccino (nome, eta_min, eta_max, efficacia, dosi_richieste,
    intervallo_somministrazione)
values ('Covidin', 12, 85, 90, 2, 50);
insert into tipo_vaccino (nome, eta_min, eta_max, efficacia, dosi_richieste,
    intervallo_somministrazione)
values ('Coronax', 45, 150, 99, 2, 40);
insert into tipo_vaccino (nome, eta_min, eta_max, efficacia, dosi_richieste)
values ('Flustop', 65, 100, 95, 1);
```

-- Lotto

```
insert into lotto
values ('cn0987', 1, 500, '2020-11-05', '2021-02-23');
insert into lotto (cod, vaccino, data_produzione, data_scadenza)
values ('1234cx', 2, '2021-01-21', '2021-04-17');
insert into lotto
values ('f4567p', 3, 500, '2021-03-13', '2021-06-30');
```

-- Appuntamento Vaccinale

```
insert into appuntamento_vaccinale (data_appuntamento, ora, centro, lotto, cittadino)
values ('2021-01-15', '09:00', 2, 'cn0987', 4);
insert into appuntamento_vaccinale (data_appuntamento, ora, centro, lotto, cittadino)
values ('2021-02-20', '18:15', 2, 'cn0987', 4);
insert into appuntamento_vaccinale (data_appuntamento, ora, centro, lotto, cittadino)
values ('2021-02-09', '15:30', 3, '1234cx', 3);
insert into appuntamento_vaccinale (data_appuntamento, ora, centro, lotto, cittadino)
values ('2021-07-1', '08:23', 2, 'cn0987', 14);
insert into appuntamento_vaccinale (data_appuntamento, ora, centro, lotto, cittadino)
values ('2021-03-03', '14:57', 5, 'cn0987', 15);
insert into appuntamento_vaccinale (data_appuntamento, ora, centro, lotto, cittadino)
values ('2021-05-01', '15:45', 5, 'cn0987', 15);
insert into appuntamento_vaccinale (data_appuntamento, ora, centro, lotto, cittadino)
values ('2021-04-30', '20:35', 1, 'cn0987', 1);
insert into appuntamento_vaccinale (data_appuntamento, ora, centro, lotto, cittadino)
values ('2021-01-7', '11:00', 1, 'f4567p', 9);
insert into appuntamento_vaccinale (data_appuntamento, ora, centro, lotto, cittadino)
values ('2021-06-15', '16:17', 4, 'cn0987', 6);
insert into appuntamento_vaccinale (data_appuntamento, ora, centro, lotto, cittadino)
values ('2021-07-15', '09:20', 4, 'cn0987', 6);
insert into appuntamento_vaccinale (data_appuntamento, ora, centro, lotto, cittadino)
values ('2021-03-03', '09:20', 2, '1234cx', 8);
```

-- Possiede Dosi

```
insert into possiede_dosi
values (1, 3, 75000);
insert into possiede_dosi
values (2, 1, 125000);
insert into possiede_dosi
values (3, 2, 9000);
insert into possiede_dosi
values (4, 1, 30000);
insert into possiede_dosi
values (5, 1, 500);
```

-- Allergia

```
insert into allergia
values ('lattosio', 1);
insert into allergia
values ('amoxicillina', 8);
insert into allergia
values ('niche', 14);
```

-- Report

insert into report (appuntamento_vaccinale, medico, centro, data_report, vaccino)

values (11, 14, 2, '2021-03-03', 2);

insert into report (appuntamento_vaccinale, medico, centro, data_report, vaccino)

values (6, 18, 5, '2021-05-01', 2);

commit;

3.3 Operazioni di cancellazione e modifica

/*

Tutte le operazioni svolte si riferiscono al database appena creato
e non sono da considerarsi sequenziali

*/

--select

-- Ogni sera viene stilato un rapporto che indica quante vaccinazioni sono state fatte nella
-- giornata in tutti i centri vaccinali suddivise per categoria di cittadino.

```
select c.tipo, count(*)  
from appuntamento_vaccinale av join cittadino c on av.cittadino = c.cod  
where av.data_appuntamento = CURRENT_DATE  
group by c.tipo;
```

-- Ogni sera, ogni centro vaccinale fa l'inventario del numero di dosi ancora
-- disponibili per ogni vaccino

```
select *  
from possiede_dosi  
order by centro, vaccino;
```

-- Ogni fine settimana, viene stilato un report che indica quante
-- vaccinazioni sono state fatte per ogni vaccino per ognuna delle categorie di
-- cittadini e quante di queste abbiano causato allergie.

```
with num_tot_vaccino as(  
    select c.tipo as tipo_cittadino, l.vaccino as tipo_vaccino, count(*) as num_vaccini  
    from appuntamento_vaccinale av join lotto l on av.lotto = l.cod join cittadino c on av.cittadino = c.cod  
    group by c.tipo, l.vaccino  
) , num_tot_report as(  
    select c.tipo as tipo_cittadino, l.vaccino as tipo_vaccino, count(*) as num_report  
    from report r join appuntamento_vaccinale av on r.appuntamento_vaccinale = av.cod  
    join cittadino c on av.cittadino = c.cod join lotto l on av.lotto = l.cod  
    where r.data_report between CURRENT_DATE - 7 and CURRENT_DATE  
    group by c.tipo, l.vaccino  
)  
select v.tipo_cittadino, v.tipo_vaccino, num_vaccini, num_report  
from num_tot_vaccino v, num_tot_report r  
where v.tipo_cittadino = r.tipo_cittadino and v.tipo_vaccino = r.tipo_vaccino;
```


--edit

```
update allergia
set cittadino = 15
where cittadino = 1;
```

```
update centro_vaccinale
set cod=6
where cod=4;
```

```
update tipo_vaccino
set cod=4
where cod=2;
```

```
update lotto
set cod='0987cx'
where cod='1234cx';
```

```
update cittadino
set cod=1000
where cod=8;
```

```
update centro_vaccinale
set cod=1234
where cod=2;
```

```
update medico
set cod=12
where cod=2;
```

--delete--

```
delete from cittadino
where cod = 1;
```

```
delete from tipo_vaccino
where cod = 3; -- non viene portato a termine perché viola vincolo in lotto
```

```
delete from cittadino
where cod=17;
-- provoca la rimozione anche di una entry da medico
```

```
delete from centro_vaccinale
where cod=2; -- non viene portata a termine perché viola un vincolo in medico
```

```
delete from lotto  
where cod='cn0987';
```

```
delete from cittadino  
where cod=8;
```

-- queste operazioni non vengono portate a termine perché i valori sono referenziati in altre tabelle