

Relazione sviluppo Base di Dati

Indice

- 1. [Introduzione](#)
- 2. [Raccolta ed analisi dei requisiti](#)
 - 2.1 [Carico di lavoro](#)
- 3. [Progettazione concettuale](#)
 - 3.1 [Vincoli di integrità](#)
- 4. [Progettazione Logica](#)
 - 4.1 [Ristrutturazione](#)
 - 4.2 [Traduzione](#)
- 5. [Normalizzazione](#)
- 6. [Progettazione Pratica](#)
 - 6.1 [Creazione del database](#)
 - 6.2 [Creazione delle tabelle](#)
 - 6.2.1 [Vaccinati](#)
 - 6.2.3 [Centri Vaccinali](#)
 - 6.2.4 [Cittadini](#)
 - 6.2.5 [Eventi Avversi](#)
 - 6.2.6 [Data set Comuni](#)
 - 6.3 [Registrazioni](#)
 - 6.3.1 [Registrare centro vaccinale](#)
 - 6.3.2 [Registrare cittadino](#)
 - 6.3.3 [Registrare vaccinato](#)
 - 6.3.4 [Inserire eventi avversi](#)
 - 6.4 [Varie operazioni](#)
 - 6.4.1 [Ottenere l'id del vaccinato](#)
 - 6.4.2 [Ottenere l'account dei cittadini](#)
 - 6.4.3 [Ottenere il riepilogo degli eventi avversi registrati da un cittadino](#)
 - 6.4.4 [Ottenere il riepilogo generale di un cittadino](#)
 - 6.4.5 [Controllare se il cittadino ha un account oppure no](#)
 - 6.4.6 [Ottenere il nome utente del cittadino](#)
 - 6.4.7 [Controllo della già avvenuta registrazione di un centro vaccinale](#)
 - 6.4.8 [Controllo della già avvenuta vaccinazione del cittadino](#)
 - 6.4.9 [Verifica dell'inserimento del corretto id in fase di registrazione](#)
 - 6.4.10 [Prospetto riassuntivo di uno specifico centro vaccinale](#)
 - 6.4.11 [Ricerca di una serie di centri vaccinali specificando il comune e la tipologia di essi](#)
 - 6.4.12 [Ricerca di una serie di centri vaccinali specificando il comune e la tipologia di essi](#)
 - 6.4.13 [Ricerca di una serie di centri vaccinali specificando una stringa rappresentante il nome \(o parte di esso\).](#)
 - 6.4.14 [Ricerca dei nomi dei centri vaccinali esistenti](#)
 - 6.4.15 [Ottenere la sigla e la provincia di un comune](#)
 - 6.4.16 [Ottenere la lista dei comuni italiani](#)

1. Introduzione

- La seguente relazione ha lo scopo di illustrare le fasi della costruzione e dello sviluppo della base dati del progetto **VaxCenter**, mostrando in modo dettagliato ogni sua componente.
- Per la realizzazione della base di dati si è deciso di procedere con il metodo sistematico della progettazione, dividendo la realizzazione del database, in una serie di sotto-fasi.
- Le sotto-fasi presenti sono:
 1. **Raccolta ed analisi dei requisiti**
 2. **Progettazione concettuale**
 3. **Progettazione Logica**
 4. **Normalizzazione**
 5. **Progettazione fisica**

Ciascuna di queste sotto-fasi produce un output sulla base di ciò che è stato ottenuto nella sotto-fase precedente, per questo, ogni sotto-fase compie, in modo intrinseco, la verifica della qualità.

La scelta di questa metodologia di lavoro permette di avere uno schema intuitivo e adatto a possibili cambiamenti, oltre al fatto che risulta di grande aiuto nella creazione della base di dati.

2. Raccolta ed analisi dei requisiti

- Analizzando le specifiche fornite per l'applicazione *VaxCenter*, i dati necessari da memorizzare sulla base di dati riguardano i **centri vaccinali**, aggiungendoli in una tabella denominata `centri_vaccinali`:

```
nomeCentro VARCHAR(30)
indirizzo
    qualificatore VARCHAR(20) [via/viale/piazza]
    nomeVia VARCHAR(30)
    numeroCivico CHAR(2)
    comune VARCHAR(30)
    siglaProvincia CHAR(2)
    cap VARCHAR(5)
tipologia VARCHAR(20) [ospedaliero, hub, aziendale]
```

Oltre alla permanenza dei centri vaccinali, è stato necessario memorizzare le informazioni dei cittadini che si sono **vaccinati**, aggiungendoli in una tabella denominata `vaccinati`:

```
nomeCentro VARCHAR(30)
nome VARCHAR(30)
cognome VARCHAR(30)
codiceFiscale CHAR(16)
dataVaccino DATE
vaxTipo VARCHAR(30) [Pfizer, AstraZeneca, Moderna, J&J]
id INTEGER
```

Per poter permettere ai cittadini di consultare le informazione di ogni centro vaccinale, è importante registrarsi al centro di propria competenza ed inserire eventuali effetti collaterali o eventi avversi collegati. Per questo motivo, è necessario memorizzare i dati dei **cittadini**, in una tabella denominata `cittadini`:

```
nome VARCHAR(30)
cognome VARCHAR(30)
nomeCentro VARCHAR(30)
dataVaccino DATE
vaxTipo VARCHAR(30) [Pfizer, AstraZeneca, Moderna, J&J]
codiceFiscale CHAR(16)
email VARCHAR(30)
username VARCHAR(30)
password VARCHAR(30)
id INTEGER
```

Nella documentazione non è specificato dove poter inserire gli eventi avversi, si è deciso di creare un ulteriore tabella dove i cittadini registrati potranno inserire gli **eventi avversi**, la tabella viene denominata `eventi_avversi` e presenta la seguente struttura:

```
id VARCHAR(50)
mal_di_testa INTEGER
febbre INTEGER
tachicardia INTEGER
dolori_muscolari INTEGER
linfadenopatia INTEGER
crisi_ipertensiva INTEGER
note VARCHAR(256)
```

In questo modo un cittadino accedendo, potrà inserire le segnalazioni opportune.

Infine si ritiene opportuno creare un ulteriore tabella dove si avranno tutti i **dati dei comuni italiani**, aggiornati al 2019, la tabella viene denominata `dataset_comuni` e presenta la seguente struttura:

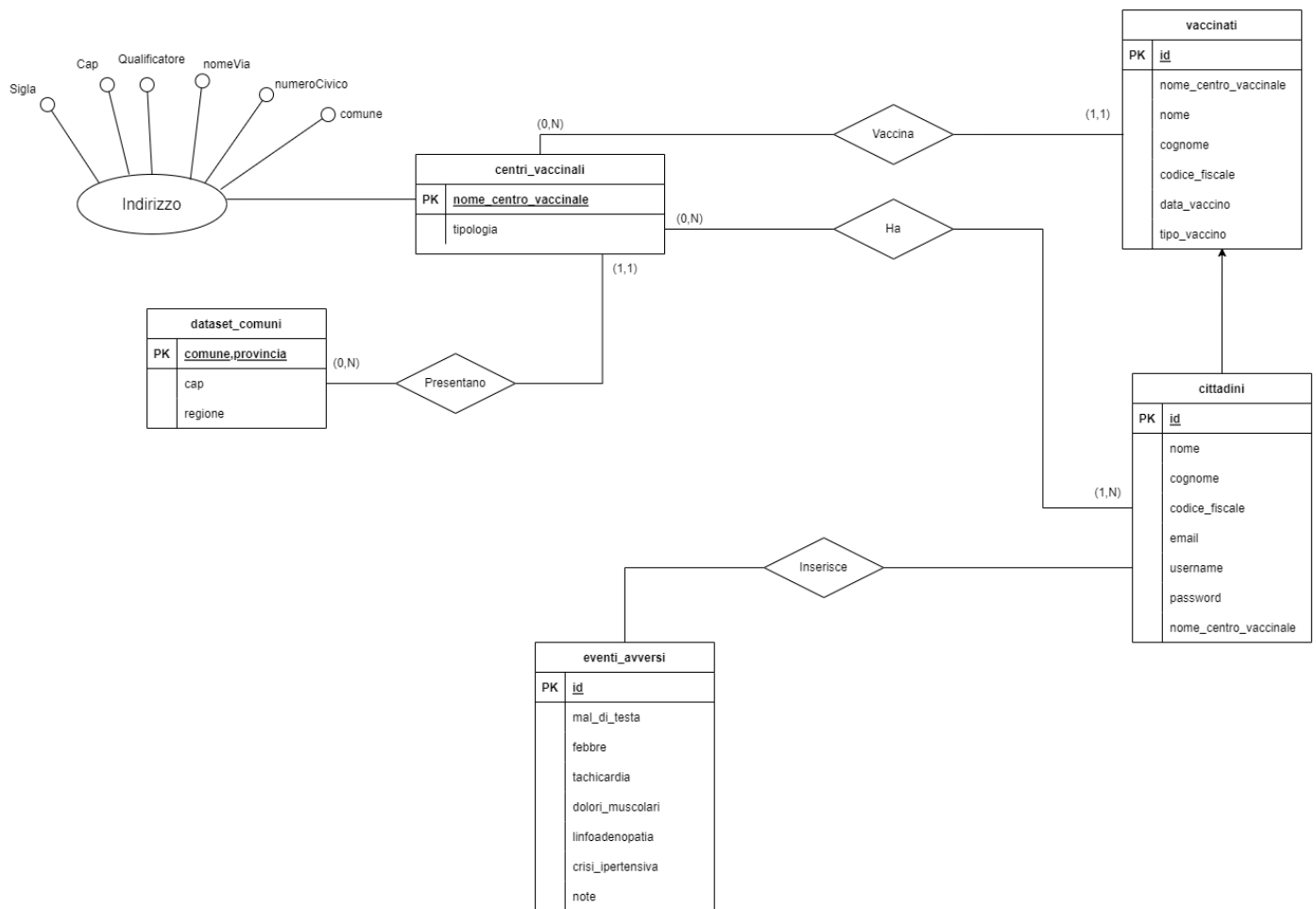
```
comune VARCHAR(40)
provincia VARCHAR(2)
cap INTEGER
regione VARCHAR(21)
```

2.1 Carico di lavoro

- Un ulteriore concetto da analizzare è il carico di lavoro, il quale è fondamentale per poter ottimizzare la base di dati;
- Si prevede che gli utenti siano un numero molto elevato, principalmente tutti i cittadini italiani, essendo esortati al vaccinarsi, e proprio per questo motivo è stato necessario analizzare le possibili operazioni da compiere su di essa;
- Attualmente in Italia, secondo **ISTAT**, i residenti sono 59.236.213, [https://www.istat.it/it/archivio/264511#:~:text=In calo il numero di,al 2019 \(-405.275 individui\);](https://www.istat.it/it/archivio/264511#:~:text=In calo il numero di,al 2019 (-405.275 individui);)
- I cittadini invece con almeno una dose di vaccino al 17/11/2022 sono 49.445.000 (<https://www.governo.it/it/cscovid19/report-vaccini/>), *si tiene conto di cittadini over 12*;
- 2353 è il numero di centri vaccinali presenti in Italia al 17/11/2022 (<https://www.governo.it/it/cscovid19/report-vaccini/>);
- Supponendo che ad ogni centro vaccinale siano registrati mediamente 17.500 cittadini, conoscendo il numero di centri vaccinali, il totale è di 41.177.500;
- il numero di vaccinazioni giornaliere è circa 31.000 al giorno [<https://lab24.ilsole24ore.com/numeri-vaccini-italia-mondo/>];
- Dopo aver analizzato i seguenti dati, si può confermare che la nostra applicazione richiede l'utilizzo di un numero elevato di dati e, inoltre, il suo uso può variare a seconda della situazione COVID. Per questo motivo immaginiamo dei picchi di utilizzo dell'applicazione in periodi caratterizzato da emergenza sanitaria.

3. Progettazione concettuale

- A partire dalla *raccolta ed analisi dei requisiti* si crea uno schema concettuale della base di dati, cioè una descrizione formale e ad alto livello del suo contenuto. Durante la progettazione concettuale vengono effettuate verifiche di qualità che possono portare a un ridisegno dello schema generato. Oltre allo schema concettuale, si stila una documentazione a corredo.
- Questa sotto-fase è fondamentale perchè permette di lavorare in maniera incrementale, ovvero concentrandosi sui dettagli sintattici e tralasciando le informazioni già richieste in precedenza. In questo modo si definisce immediatamente lo schema logico. Inoltre, tenendo conto della "rigidità" dei modelli ottenuti, senza questa sotto-fase si rischierebbe di omettere delle informazioni, complicando successivamente le procedure di modifica.
- **Essere bravi e concentrati in questa fase, porterà benefici nelle fasi successive**



3.1 Vincoli di integrità

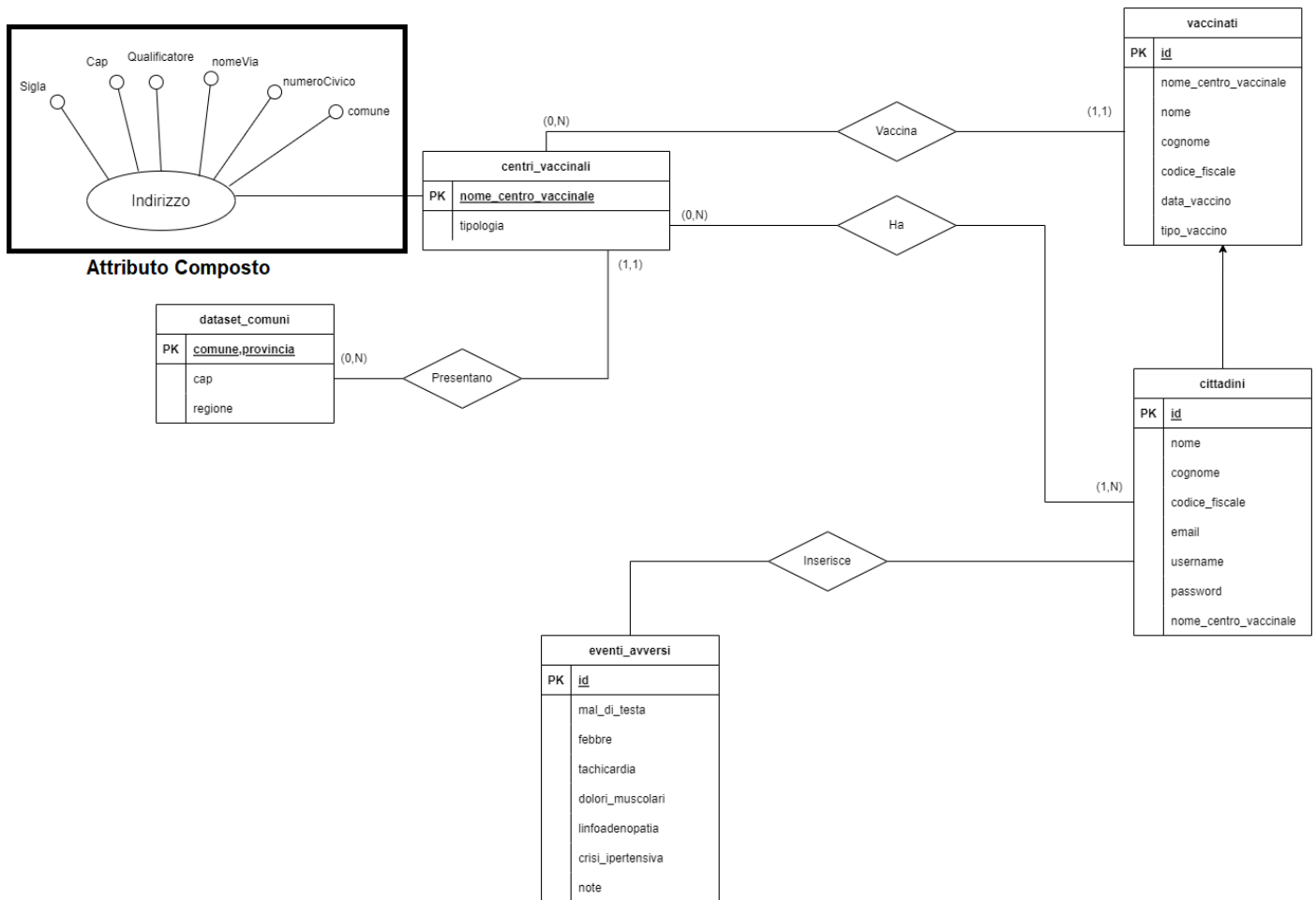
- **V1:** Il qualificatore può essere via, viale, piazza
- **V2:** La tipologia deve essere ospedaliero, hub, aziendale
- **V3:** il vaccino deve essere Pfizer, AstraZeneca, Moderna, J&J
- **V4:** Un cittadino registrato non può registrarsi nuovamente nello stesso centro vaccinale
- **V5:** Un cittadino registrato deve essere per forza vaccinato
- **V6:** Un operatore aggiunge un cittadino vaccinato alla volta
- **V7:** Gli eventi avversi rilevati sono mal di testa, febbre, dolori muscolari e articolari, linfadenopatia, tachicardia e crisi ipertensiva.

4. Progettazione Logica

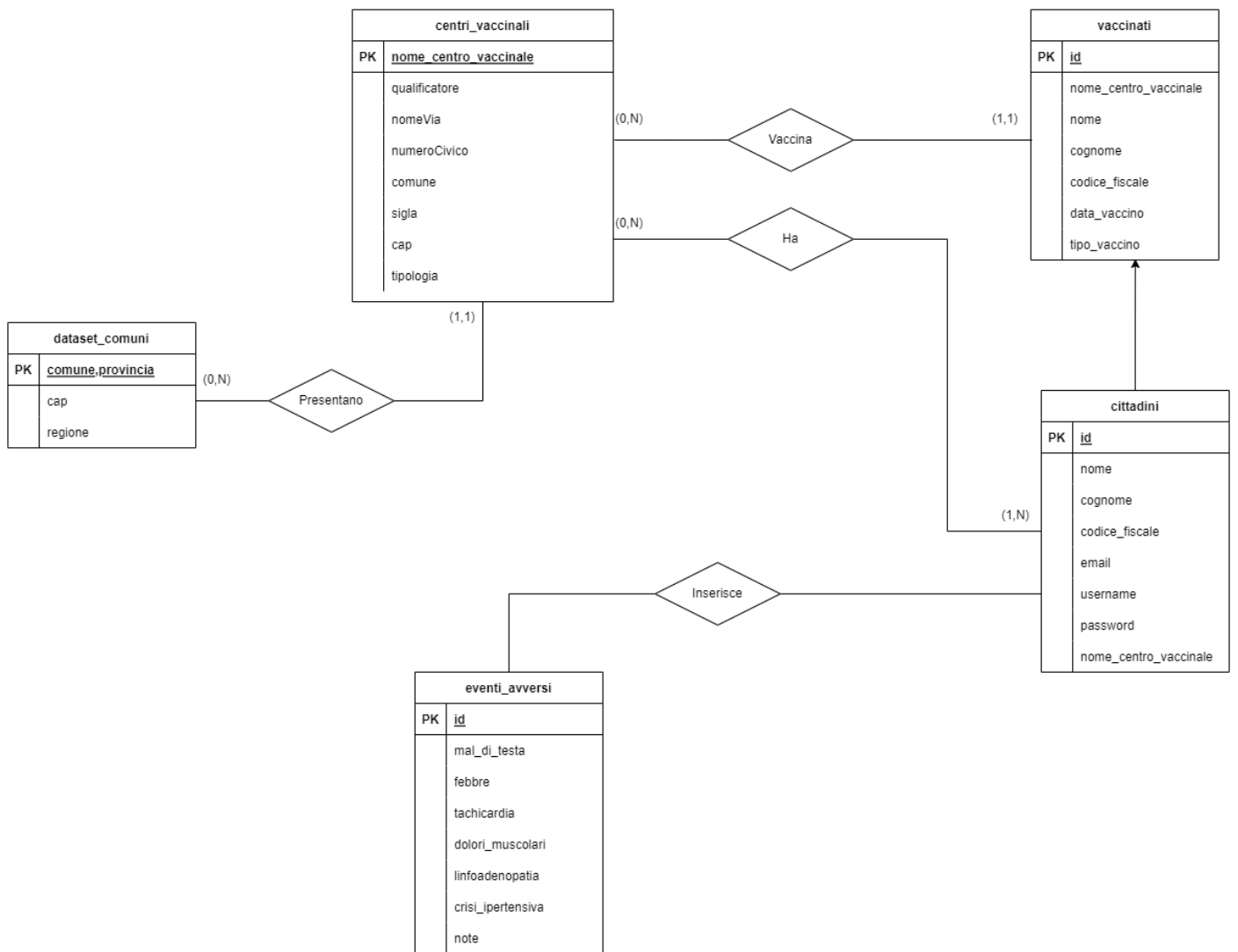
- Dopo aver ottenuto lo schema della progettazione concettuale, che è indipendente da un particolare sistema di gestione dati, si produce uno schema logico specifico per un particolare **DBMS**
- Per procedere in maniera standard, si scompone questa fase in due sotto-fasi:
 - **Ristrutturazione**
 - **Traduzione**

4.1 Ristrutturazione

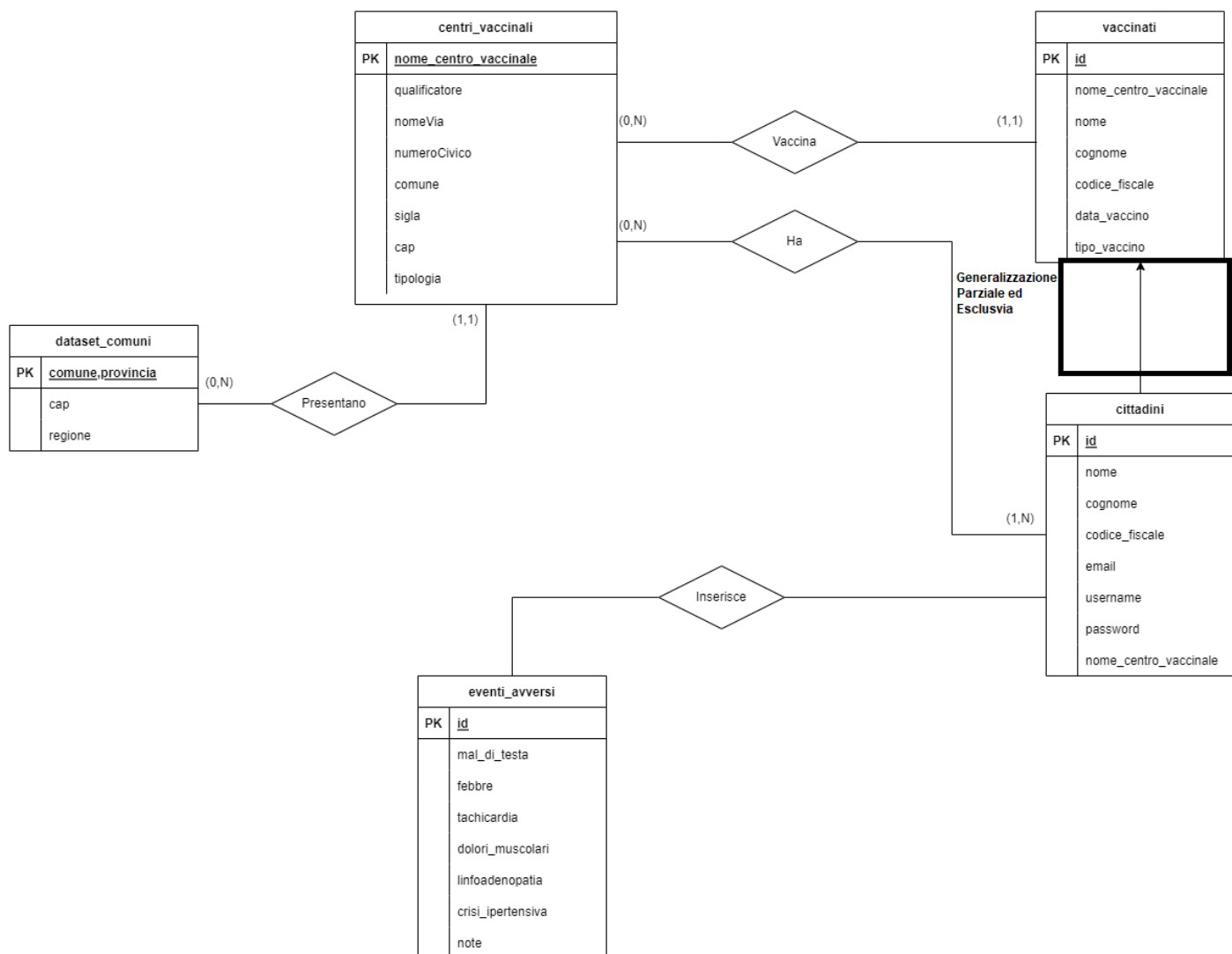
- Si ottiene uno schema ER semplificato, equivalente a quello di partenza, ma con l'eliminazione dei costrutti non direttamente rappresentabili nel modello relazione; come ad esempio la generalizzazione e gli attributi composti;
- Questa sotto-fase non è "univoca", ma dipende dal carico di lavoro e dalle considerazioni rilevate;
- Analizzando il risultato dello schema concettuale, si nota la presenza di un attributo composto: Indirizzo



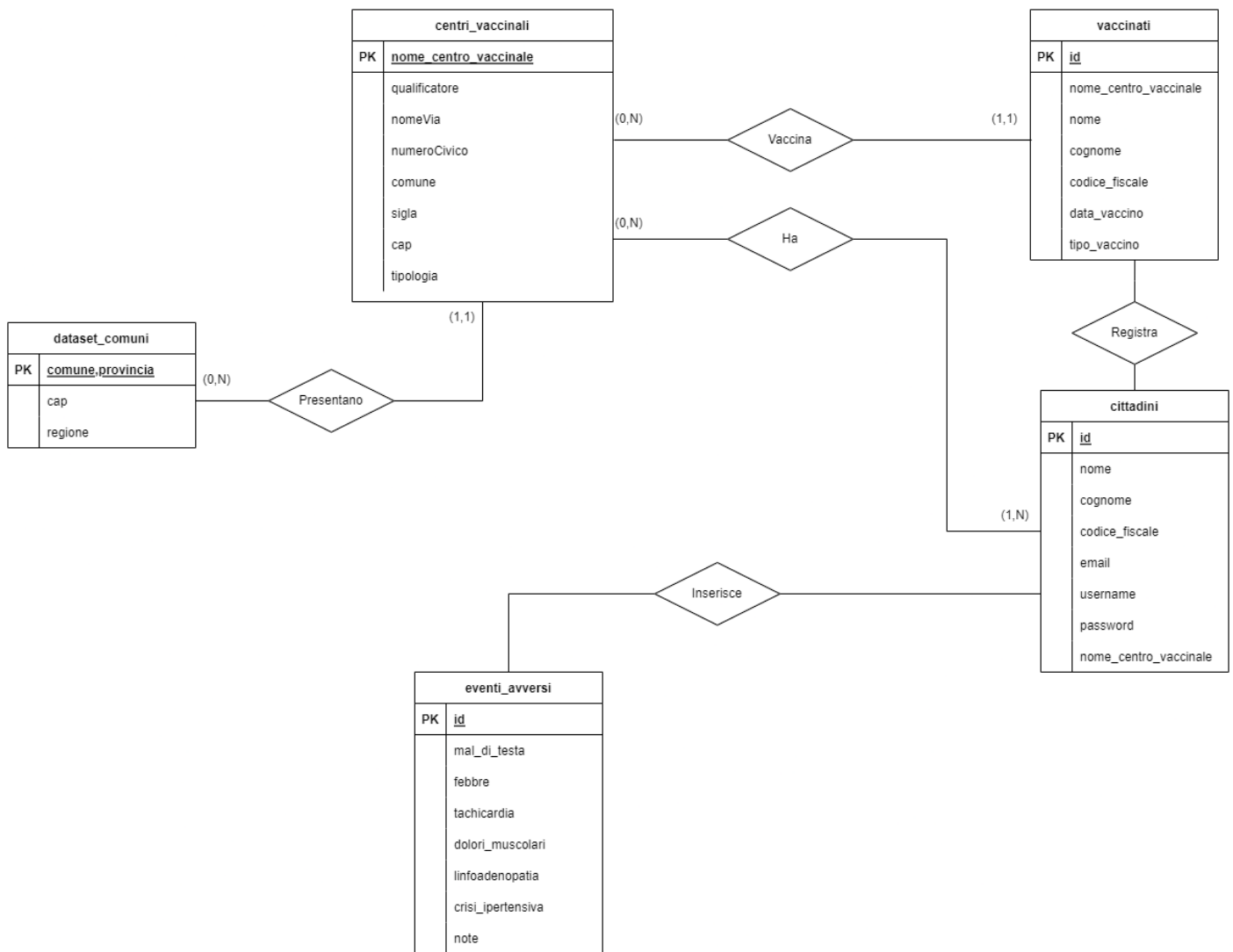
- Per rendere il carico di lavoro meno oneroso e contestualmente più agevoli le possibili interrogazioni che si potrebbero effettuare, gli elementi dell'attributo composto vengono scomposti ed inseriti nella tabella *centri_vaccinali*.
- Successivamente per rimuovere l'attributo composto Indirizzo, si integra la tabella *centri_vaccinali* con l'informazione degli attributi di Indirizzo. La figura seguente mostra la tabella *centri_vaccinali* dopo la rimozione dell'indirizzo composto:



- Un ulteriore costrutto da rimuovere è quello di **generalizzazione**, in quanto non rappresentabile direttamente nel modello relazionale. Nello specifico si tratta di una **generalizzazione parziale ed esclusiva**.



- Per eliminare questa generalizzazione, si utilizza la sostituzione con associazioni essendo la soluzione più generale e sempre applicabile, oltre che permette di risparmiare memoria utilizzata.



4.2 Traduzione

- A partire dallo schema ER ristrutturato, si ottiene uno schema equivalente relazionale;
- Le entità vengono tradotte in relazioni, mentre le associazioni sono tradotte in relazioni o chiavi esterne;
- Prima di tutto, si traducono le entità in relazioni:

centri_vaccinale(nome_centro_vaccinale,qualificatore,nomeVia,numeroCivico,comune,sigla, cap,tipologia)

vaccinati(id,nome_centro_vaccinale,nome,cognome,codice_fiscale, data_vaccino,tipovaccino)

cittadini(id,nome,cognome,codice_fiscale,email,username,password,nome_centro_vaccinale)

eventi_avversi(id,mal_di_testa,febbre, tachicardia,dolori_muscolari, linfadenopatia,crisi_ipertensiva,note₀)

dataset_comuni(comune,provincia,cap,regione)

- Ai fini di efficienza e per evitare la creazione di ulteriori tabelle, si è deciso di rappresentare tutte le relazioni con chiavi esterne, nello specifico :
 - *Presentano*: viene rappresentata con l'utilizzo di **comune e provincia**
 - *Vaccina*: viene rappresentata con l'utilizzo di **nome_centro_vaccinale**;
 - *Registra*: viene rappresentata con l'utilizzo dell' **id**
 - *Ha* : viene rappresentata con l'utilizzo di **nome_centro_vaccinale**;
 - *Inserisce*: viene rappresentata con l'utilizzo dell' **id**

5. Normalizzazione

- Data la ridotta complessità della struttura del database, non è stato ritenuto opportuno trattare nel dettaglio la sotto-fase di normalizzazione.

6. Progettazione Pratica

6.1. Creazione del database

```
CREATE DATABASE vaxcenter;
```

6.2. Creazione delle tabelle

6.2.1 Vaccinati

```
CREATE TABLE vaccinati(  
    id VARCHAR(16) PRIMARY KEY,  
    nome_centro_vaccinale VARCHAR(80) REFERENCES centri_vaccinali,  
    nome VARCHAR(50),  
    cognome VARCHAR(50),  
    codice_fiscale VARCHAR(50),  
    data_vaccino VARCHAR(40),  
    tipo_vaccino VARCHAR(50));
```

id	nome	cognome	codice_fiscale	data_vaccino	tipo_vaccino	nome_centro_vaccinale
----	------	---------	----------------	--------------	--------------	-----------------------

6.2.2 Centri Vaccinali

```
CREATE TABLE centri_vaccinali(  
    nome_centro_vaccinale VARCHAR(50) PRIMARY KEY,  
    qualificatore VARCHAR(7),  
    nome_via VARCHAR(30),  
    civico VARCHAR(6),  
    provincia CHAR(2),  
    comune VARCHAR(30),  
    cap INTEGER,  
    tipologia VARCHAR(20)  
    FOREIGN KEY(comune,provincia) REFERENCES dataset_comuni);
```

nome_centro_vaccinale	qualificatore	nome_via	civico	provincia	cap	tipologia
-----------------------	---------------	----------	--------	-----------	-----	-----------

6.2.3 Cittadini

```
CREATE TABLE cittadini(  
    id VARCHAR(16) PRIMARY KEY REFERENCES vaccinati,  
    nome VARCHAR(50),  
    cognome VARCHAR(50),  
    codice_fiscale VARCHAR(50),  
    email VARCHAR(50),  
    username VARCHAR(50),  
    password VARCHAR(150),  
    nome_centro_vaccinale VARCHAR(50) REFERENCES centri_vaccinali);
```

id	nome	cognome	codice_fiscale	email	username	password	nome_centro_vaccinale
----	------	---------	----------------	-------	----------	----------	-----------------------

6.2.4 Eventi Avversi

```
CREATE TABLE eventi_avversi(  
    id VARCHAR(16) PRIMARY KEY REFERENCES cittadini,  
    mal_di_testa INTEGER,  
    febbre INTEGER,  
    tachicardia INTEGER,  
    dolori_muscolari INTEGER,  
    linfadenopatia INTEGER,  
    crisi_ipertensiva INTEGER,  
    note VARCHAR(256));
```

id	mal_di_testa	febbre	tachicardia	dolori_muscolari	linfadenopatia	crisi_ipertensiva	note
----	--------------	--------	-------------	------------------	----------------	-------------------	------

6.2.5 Data set Comuni

```
CREATE TABLE dataset_comuni(  
    comune VARCHAR(40),  
    provincia VARCHAR(2),  
    cap INTEGER,  
    regione VARCHAR(21)
```

```
PRIMARY KEY (comune,provincia)
);
```

comune	provincia	cap	regione
--------	-----------	-----	---------

6.3. Registrazioni

6.3.1 Registrare centro vaccinale

```
INSERT INTO centri_vaccinali (nome_centro_vaccinale, qualificatore, nome_via, civico, provincia, comune, cap, tipologia)
VALUES (?, ?, ?, ?, ?, ?, ?, ?);
```

nome_centro_vaccinale	qualificatore	nome_via	civico	provincia	comune	cap	tipologia
MALPENSA FIERE	VIA	XI SETTEMBRE	16	VA	BUSTO ARSIZIO	21050	HUB

6.3.2 Registrare cittadino

```
INSERT INTO cittadini (id, nome, cognome, codice_fiscale, email, username, password, nome_centro_vaccinale)
VALUES (?, ?, ?, ?, ?, ?, ?, ?)
```

id	nome	cognome	codice_fiscale	email	username	password	nome_centro_vaccinale
00000000...	MARIO	ROSSI	RSSMRA78M14L319P	MROSSI@GMAIL.COM	MARIOR	MROSSI15!	MALPENSA FIERE

6.3.3 Registrare vaccinato

```
INSERT INTO vaccinati (id, nome_centro_vaccinale, nome, cognome, codice_fiscale, data_vaccino, tipo_vaccino)
VALUES (?, ?, ?, ?, ?, ?, ?)
```

id	nome	cognome	codice_fiscale	data_vaccino	tipo_vaccino	nome_centro_vaccinale
000000000000000000	MARIO	ROSSI	RSSMRA78M14L319P	14-11-1978	PFIZER	MALPENSA FIERE

6.3.4 Inserire eventi avversi

```
INSERT INTO
eventi_avversi (id, mal_di_testa, febbre, tachicardia, dolori_muscolari, linfadenopatia, crisi_ipertensiva, note)
VALUES (?, ?, ?, ?, ?, ?, ?, ?)
```

username	mal_di_testa	febbre	tachicardia	dolori_muscolari	linfadenopatia	crisi_ipertensiva	note
000000000000000000	4	0	1	1	2	3	MAL DI PANCIA

6.4. Varie operazioni

- 6.4.1 Ottenere l'id del vaccinato

```
SELECT id FROM vaccinati
```

- 6.4.2 Ottenere l'account dei cittadini

```
SELECT * FROM eventi_avversi WHERE id = ?
```

- 6.4.3 Ottenere il riepilogo degli eventi avversi registrati da un cittadino

```
SELECT * FROM eventi_avversi WHERE id = ?
```

- 6.4.4 Ottenere il riepilogo generale di un cittadino

```
SELECT * FROM cittadini WHERE username = ?
```

- 6.4.5 Controllare se il cittadino ha un account oppure no

```
SELECT *
FROM cittadini
WHERE username = encrypt(account.getUserId(), SECRETKEY) AND password = sha256(account.getPassword())
```

- 6.4.6 Ottenere il nome utente del cittadino

```
SELECT * FROM cittadini WHERE username = ?
```

- 6.4.7 Controllo della già avvenuta registrazione di un centro vaccinale

```
SELECT * FROM centri_vaccinali WHERE nome_centro_vaccinale = ?
```

- 6.4.8 Controllo della già avvenuta vaccinazione del cittadino

```
SELECT * FROM vaccinati WHERE codice_fiscale = ?
```

- **6.4.9 Verifica dell'inserimento del corretto id in fase di registrazione**

```
SELECT id FROM vaccinati WHERE codice_fiscale = ?
```

- **6.4.10 Prospetto riassuntivo di uno specifico centro vaccinale**

```
SELECT COUNT(mal_di_testa) AS segnalazioni_mdt, AVG(mal_di_testa) AS media_mdt,
COUNT(febbre) AS segnalazioni_febbre, AVG(febbre) AS media_febbre,
COUNT(tachicardia) AS segnalazioni_tachicardia, AVG(tachicardia) AS media_tachicardia,
COUNT(dolori_muscolari) AS segnalazioni_dm, AVG(dolori_muscolari) AS media_dm,
COUNT(linfoadenopatia) AS segnalazioni_linfoadenopatia, AVG(linfoadenopatia) AS media_linfoadenopatia,
COUNT(crisi_ipertensiva) AS segnalazioni_ci, AVG(crisi_ipertensiva) AS media_ci
FROM eventi_avversi JOIN cittadini USING (username)
WHERE nome_centro_vaccinale = nomeCentroVaccinale
```

- **6.4.11 Ricerca di una serie di centri vaccinali specificando il comune e la tipologia di essi**

```
SELECT *
FROM centri_vaccinali
WHERE comune LIKE comune.toUpperCase() AND tipologia LIKE tipologia.toString().toUpperCase()
```

- **6.4.12 Ricerca di una serie di centri vaccinali specificando il comune e la tipologia di essi**

```
SELECT *
FROM centri_vaccinali
WHERE nome_centro_vaccinale LIKE nome.toUpperCase()
```

- **6.4.13 Ricerca di una serie di centri vaccinali specificando una stringa rappresentante il nome (o parte di esso)**

```
SELECT * FROM centri_vaccinali
```

- **6.4.14 Ricerca dei nomi dei centri vaccinali esistenti**

```
SELECT nome_centro_vaccinale FROM centri_vaccinali
```

- **6.4.15 Ottenere la sigla e la provincia di un comune**

```
SELECT provincia,cap FROM dataset_comuni WHERE comune = ?
```

- **6.4.16 Ottenere la lista dei comuni italiani**

```
SELECT comune FROM dataset_comuni
```